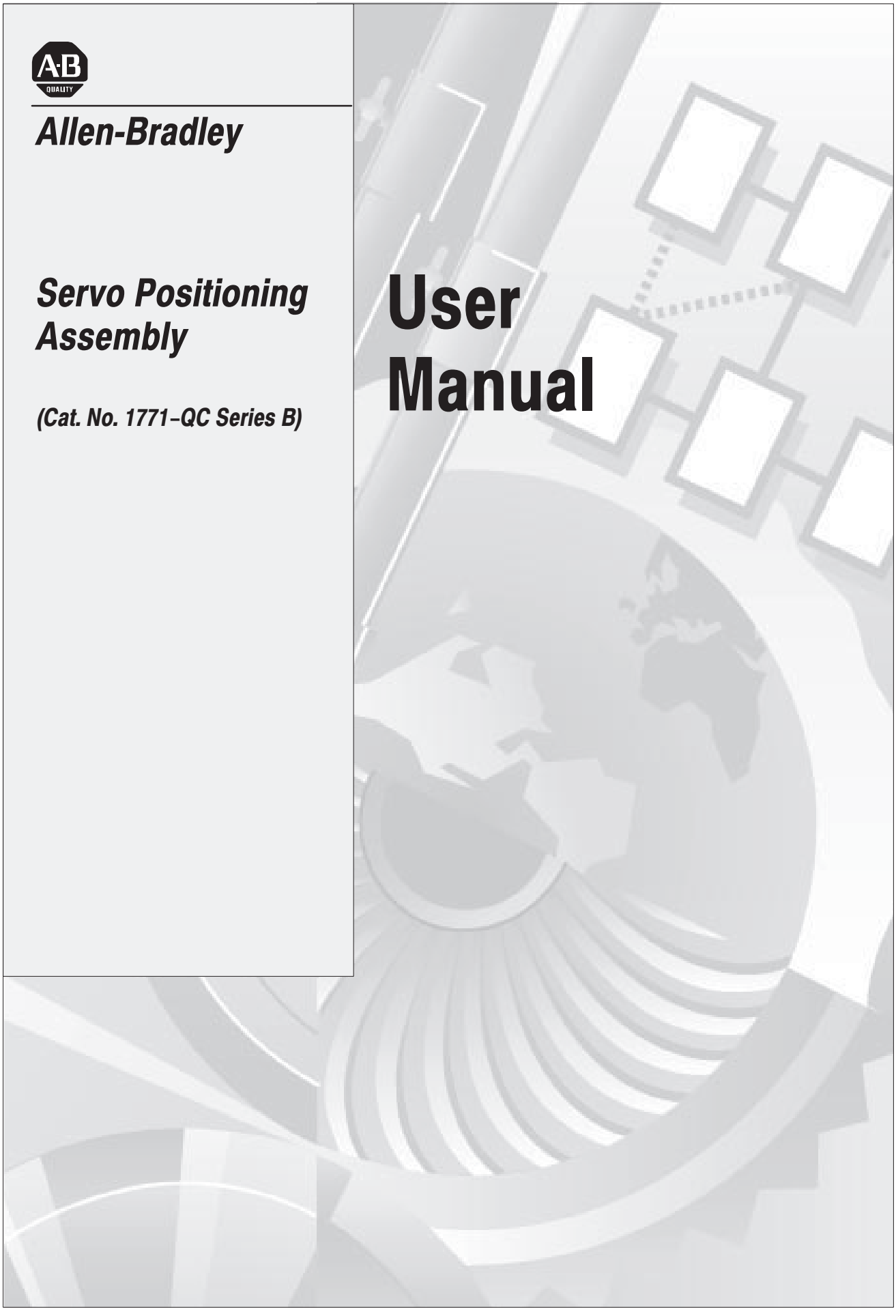


Allen-Bradley

Servo Positioning Assembly

(Cat. No. 1771–QC Series B)

User Manual

Table of Contents

Using This Manual 1-1

Manual's Purpose 1-1
Audience 1-1
Vocabulary 1-1
Manual Organization 1-2

Introducing the Servo Positioning Assembly 2-1

Chapter Objectives 2-1
What is the Servo Positioning Assembly? 2-1
Its Applications 2-2
Its Function 2-2
Its Features 2-4
Summary 2-7

Positioning Concepts 3-1

Chapter Objectives 3-1
Closed-Loop Positioning 3-1
Leadscrew Pitch 3-6
Encoder Feedback 3-7
Summary 3-11

Positioning With an Allen-Bradley Programmable Controller 4-1

Chapter Objectives 4-1
Where the Servo Positioning Assembly Fits In 4-1
Independent of I/O Scan 4-2
Move/Moveset 4-2
In Position 4-8
Synchronizing Axes 4-8
Specifying Axis Position 4-10
Summary 4-13

Hardware Description 5-1

Chapter Objectives 5-1
Indicators 5-1
Inputs/Outputs 5-2
External Power Supplies 5-7
Compatible Processors 5-8
Fault Responses 5-9
Specifications 5-11
Summary 5-13

Installing the Assembly 6-1

Chapter Objectives 6-1
Configuring the Modules 6-1
Setting Switches and Jumpers 6-3
Keying 6-7
Inserting the Module 6-9
Connecting to Terminals 6-10
Connecting A-B Encoder and Drive 6-27
Start-up Sequence 6-31
Summary 6-32

Formatting and Interpreting Data Blocks 7-1

Chapter Objectives 7-1
Relationship of Data Blocks 7-1
Status Block 7-4
Parameter Block 7-17
Moveset Block 7-40
Command Block 7-60
Summary 7-79

Programming 8-1

Chapter Objectives 8-1
Programming Objectives 8-1
PLC-2 Family Block Transfer Instructions 8-3
PLC-2-Family Block Transfer Timing 8-6
PLC-3 Block Transfer Instructions 8-13
PLC-3 Block Transfer Timing 8-14
Programming Example 8-21
Summary 8-40

Integrating Axes 9-1

Chapter Objectives 9-1
Open-Loop Procedure 9-1
Closed-Loop Procedure 9-6
Tachometer Calibration 9-8
Summary 9-11

Troubleshooting 10-1

Chapter Objectives 10-1
Monitoring 1771-M3 Controller Indicators 10-1
Monitoring 1771-ES Expander Indicators 10-3
Monitoring the Status Block 10-4
Troubleshooting Flowchart 10-7
Summary 10-12

Glossary A-1

Status Block B-1

Parameter Block C-1

Moveset Block 6-1

Command Block E-1

Chapter 1

Using This Manual

Manual's Purpose

This manual shows you how to use the series B Servo Positioning Assembly (cat. no. 1771-QC). If you have a series A Servo Positioning Assembly, refer to publication 1771-817.

Audience

To use the servo positioning assembly, you must be able to program and operate an Allen-Bradley PC processor. In particular, you must be able to program block transfer instructions.

In this manual, we assume that you know how to do this. If you don't, refer to the appropriate manual for the PC processor you will be using. Consult our Publication Index (publication SD499) for a list of our publications.

Vocabulary

Some inconsistency exists throughout industry in the nomenclature used for components of closed-loop servo positioning systems. Therefore, as you read this manual, you should be aware of the names we use for these components.

- We refer to the Servo Controller Module (cat. no. 1771-M3) as the 1771-M3 controller.
- We refer to the Servo Expander Module (cat. no. 1771-ES) as the 1771-ES expander.
- We refer to the device that receives the velocity command signal from the 1771-ES expander as the servo drive. The servo drive converts ac power to dc power for the servo motor in proportion to the velocity command signal. What we refer to here as the servo drive, others may refer to as a servo controller. So, if you refer to this device as a servo controller, be aware of our nomenclature as you read this manual.
- PC refers to programmable controller.

For an extensive list of terms we use this publication, refer to the glossary in appendix A.

Chapter 1
Troubleshooting

Manual Organization

This manual is organized into the following chapters:

Chapter	Title	What's Covered
2	Introducing the Servo Positioning Assembly	an overview of the servo positioning assembly, its applications, functions, and features
3	Positioning Concepts	concepts of closed-loop positioning, including velocity loop, positioning loop, and feed forward
4	Positioning with Allen-Bradley PC's	the servo positioning assembly's position in a servo system, and the servo positioning assembly's communication with the PC processor
5	Describing Hardware	describing the servo positioning assembly, its specifications, and its compatibility with other hardware components you will need for a closed-loop positioning system
6	Installing the Assembly	installing the servo positioning assembly and interconnecting hardware
7	Formatting and Interpreting Data Blocks	formatting parameter, move description, and control data for block transfer to the servo positioning assembly-interpreting status and diagnostic data received in block transfer from the servo positioning assembly
8	Programming	generating a ladder-diagram program to transfer data blocks between the PC data table and the servo positioning assembly
9	Integrating Axes	adjusting the servo positioning assembly for optimum operation with the machine axis it is to control
10	Troubleshooting	using indicator status and status-block information to diagnose and correct problems

Chapter 2

Introducing the Servo Positioning Assembly

Chapter Objectives

This chapter gives you an overview of the servo positioning assembly, its applications, functions and features.

What is the Servo Positioning Assembly?

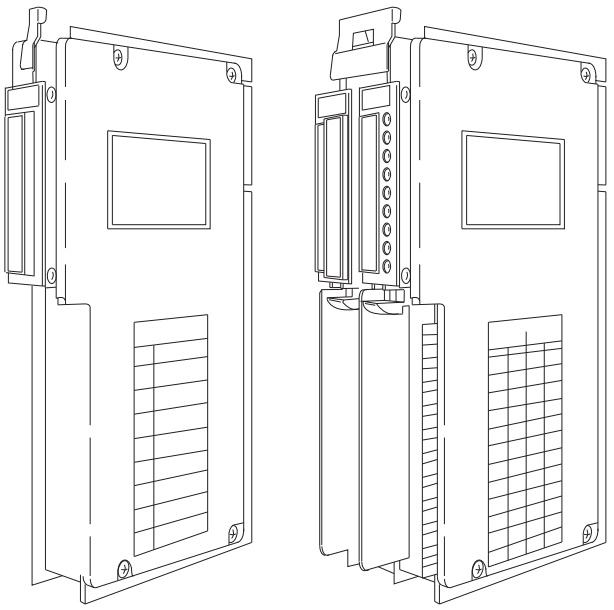
A servo positioning assembly controls the motion of one of your axes. It consists of:

- one Servo Controller Module (cat. no.1771-M3)
- one Servo Expander Module (cat. no. 1771-ES) that includes two Field Wiring Arms (cat. no. 1771-WB)

With a basic servo positioning assembly (plus a servo drive, motor, tachometer, and encoder) you can control the motion of one user-supplied machine axis. You can add a second 1771-ES expander to control a second axis and a third 1771-ES expander to control a third axis. A 1771 I/O chassis can accommodate one 1771-M3 controller and a maximum of three 1771-ES expanders.

The 1771-M3 controller requires one I/O chassis slot; it requires no wiring (figure 2.1a). You can install it at any I/O slot in the I/O chassis. The 1771-ES expander requires a pair of slots that make up an I/O module group (Figure 2.1b). You make all wiring connections to the 1771-ES expander.

Figure 2.1
Servo Positioning Assembly



(a) Servo Controller Module (cat. no. 1771 – M3)

(b) Servo expander Module (cat. no. 1771 – ES)

17954

Its Applications

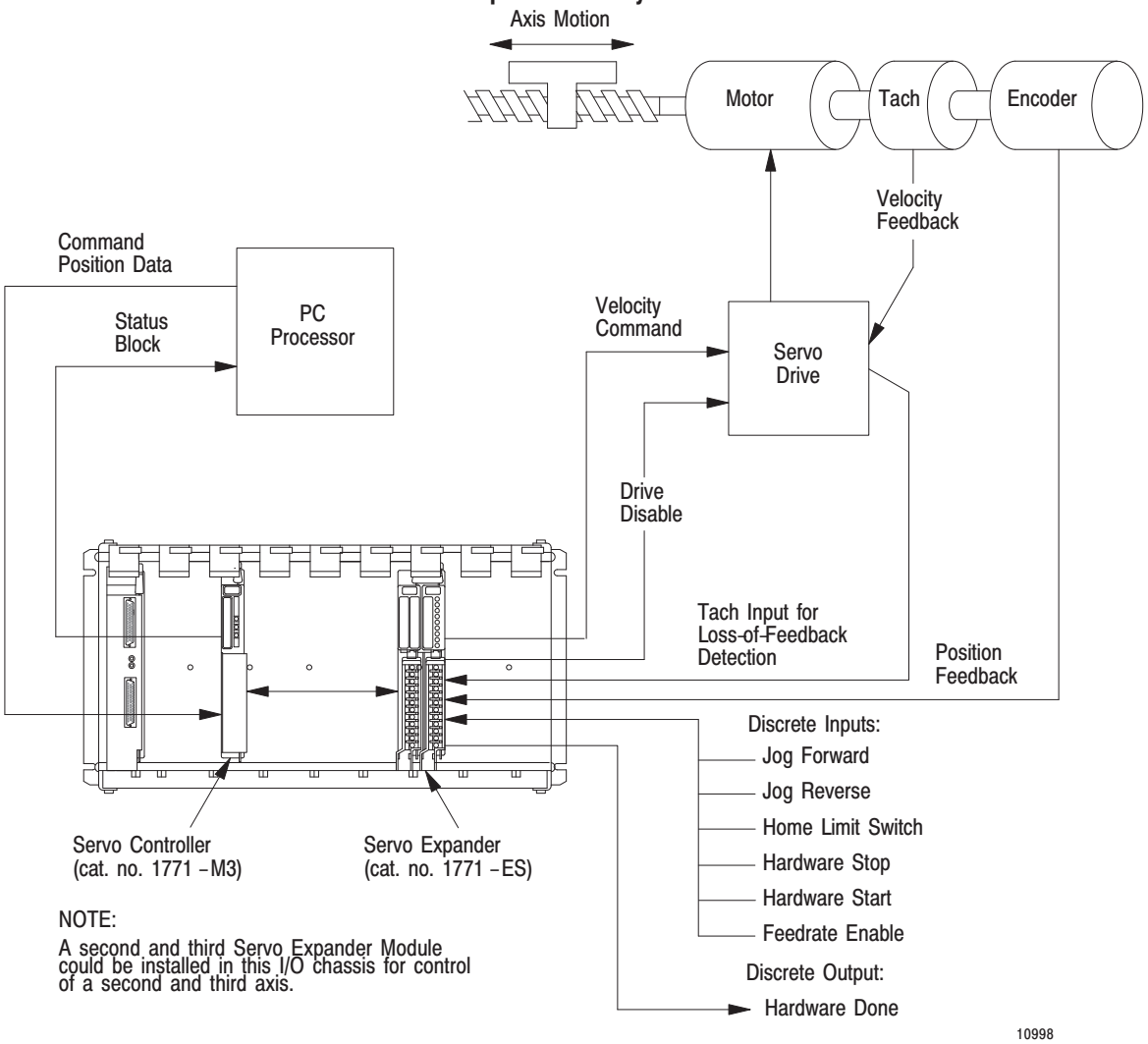
Typical applications for a servo positioning assembly include positioning for:

- grinding
- transfer lines
- material handling
- drilling
- riveting
- rotary indexing
- v-belt cutting
- glass cutting

Its Function

Figure 2.2 shows a servo system for closed-loop axis control. The 1771-M3 controller communicates with the 1771-ES expander through I/O chassis backplane connections.

Figure 2.2
Closed-loop Axis Servo System



The PC processor sends commands and user-programmed data from the data table to the 1771-M3 controller as directed by a block-transfer write instruction. The 1771-M3 controller coordinates the block transfer automatically, keeping ladder diagram programming to a minimum.

Based on information it receives from the processor, the 1771-M3 controller sends axis motion commands to the 1771-ES expander.

The 1771-ES expander closes the servo positioning loop. It commands axis motion by generating an analog voltage for your servo drive. Every 2.4 milliseconds (ms) it updates this analog output voltage according to motion commands from the 1771-M3 controller, discrete inputs, and

feedback from your encoder. The 1771-ES expander is able to provide this fast servo sample rate because the update is independent of the I/O scan.

A drive-disable output provides a signal to disable the servo drive in conditions such as loss-of-feedback or a hardware-stop signal. A hardware-done output signals the completion of each single-step move. Discrete hardware inputs include:

- hardware stop
- jog forward
- jog reverse
- home limit
- feedrate enable
- hardware start

The 1771-M3 controller sends axis status and diagnostic data to the data table as directed by a block-transfer read instruction. Because axis-command and status data is stored in the data table, axis motion control can interact with other axes, discrete I/O, and report generation.

Its Features

See the following table for a list of the many useful benefits you'ss derive from an A-B servo positioning assembly.

Feature	Benefit
incremental digital encoder feedback	precise closed-loop positioning
absolute or incremental positioning commands	programming flexibility
programmable gain break	precise positioning at low speed with stability at high speed
programmable acceleration/deceleration	optimize the machine cycle time over varying loads
programmable in-position band	flexible positioning accuracy
programmable jog rates	flexible manual positioning
programmable dwell	precise dwell times
excess-following-detection	automatic drive shutdown if the axis following error becomes too large
loss-of-feedback detection	allow automatic drive shutdown during a move if tachometer or encoder feedback is lost
software travel limits	guards against axis overtravel
backlash takeup	compensates for mechanical backlash
offset	compensates for a variation in tool length or fixture dimension
preset	easy redefinition of axis coordinates

Feature	Benefit
optically isolated analog output[1]	guards against noise entering the backplane circuits and limits the potential for damage due to improper connection
external hardware start[1]	synchronizes moves with other axes
encoder input selectable for high-true or low-true[1]	compatibility with a wider range of encoders
synchronized start of feedrate override[1]	activates a pre-loaded feedrate override value to change speed on several axes simultaneously
sensing of customer power supply loss[1]	an orderly shutdown of the servo system and to provide you with this diagnostic information
feed forwarding[1]	to allow you to reduce following error by up to 99.9% without increasing instability
constant-velocity command[1]	runs an axis continuously at a selected velocity (could apply to controlling a conveyor with no programmed end point)
moveset override[1]	Modifies a moveset while it is being executed
diagnostic words in the status block[1]	provide your ladder-diagram program with access to diagnostic information
for	hardware and program troubleshooting

[1]These features are only available on the series B servo positioning assembly.

Summary

This chapter was intended to be very general. Upcoming chapters cover these topics in greater detail. To prepare for those details, read about positioning concepts in chapter 3.

Chapter 3

Positioning Concepts

Chapter Objectives

This chapter presents positioning concepts and terminology. If you are thoroughly familiar with the concepts of closed-loop servo positioning, you can skip ahead to chapter 4.

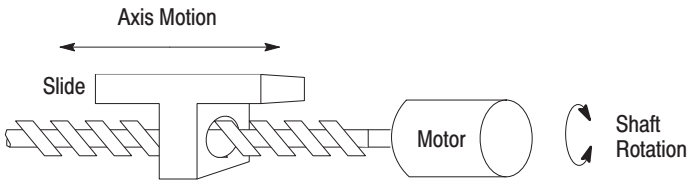
Closed-Loop Positioning

Closed-loop positioning is a precise means of moving an object from one position to another. Typically, an electric motor supplies the mechanical power, and the needed motion is linear. Therefore, we must convert the rotary motion of the motor's shaft to linear motion.

Axis Motion

One common method of converting rotary motion to linear motion is with a leadscrew (Figure 3.1)

Figure 3.1
Leadscrew Converting Rotory Motor Motion Into Linear Axis Motion



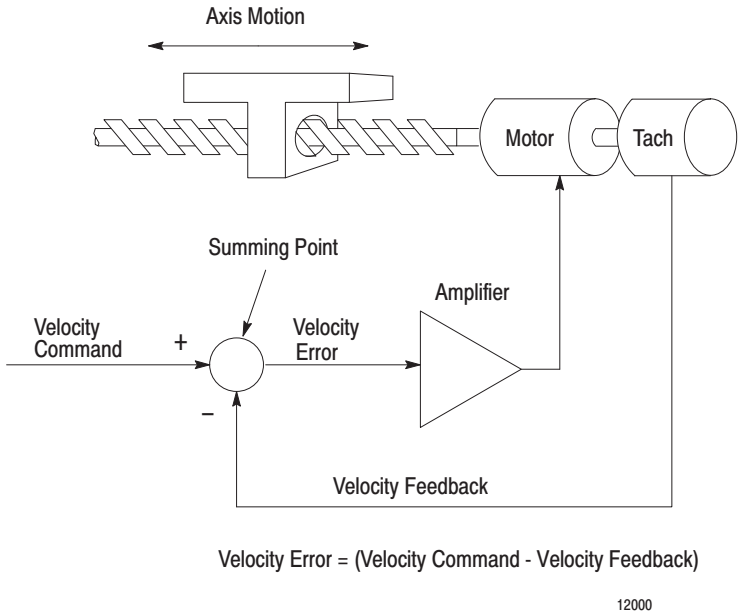
The leadscrew assembly is referred to as the axis. A leadscrew assembly consists of a long threaded shaft (the leadscrew) and slide having an internal thread that matches the leadscrew. When the motor rotates the leadscrew clockwise, the slide moves forward. When the motor rotates the leadscrew counterclockwise, the slide moves backward.

Velocity Loop

Most closed-loop servo positioning installations use a dc motor to power the leadscrew. To accurately control the velocity of the dc motor, we need a velocity loop (Figure 3.2).

The velocity loop contains a summing point, an amplifier, and a tachometer. A tachometer is a precision generator that produces a voltage signal directly proportional to the angular velocity of the motor shaft. The output of the tachometer is the velocity feedback signal which is subtracted from the velocity command signal. The difference is the velocity error signal that is amplified to provide power for the motor to run at the commanded velocity.

Figure 3.2
Velocity Loop

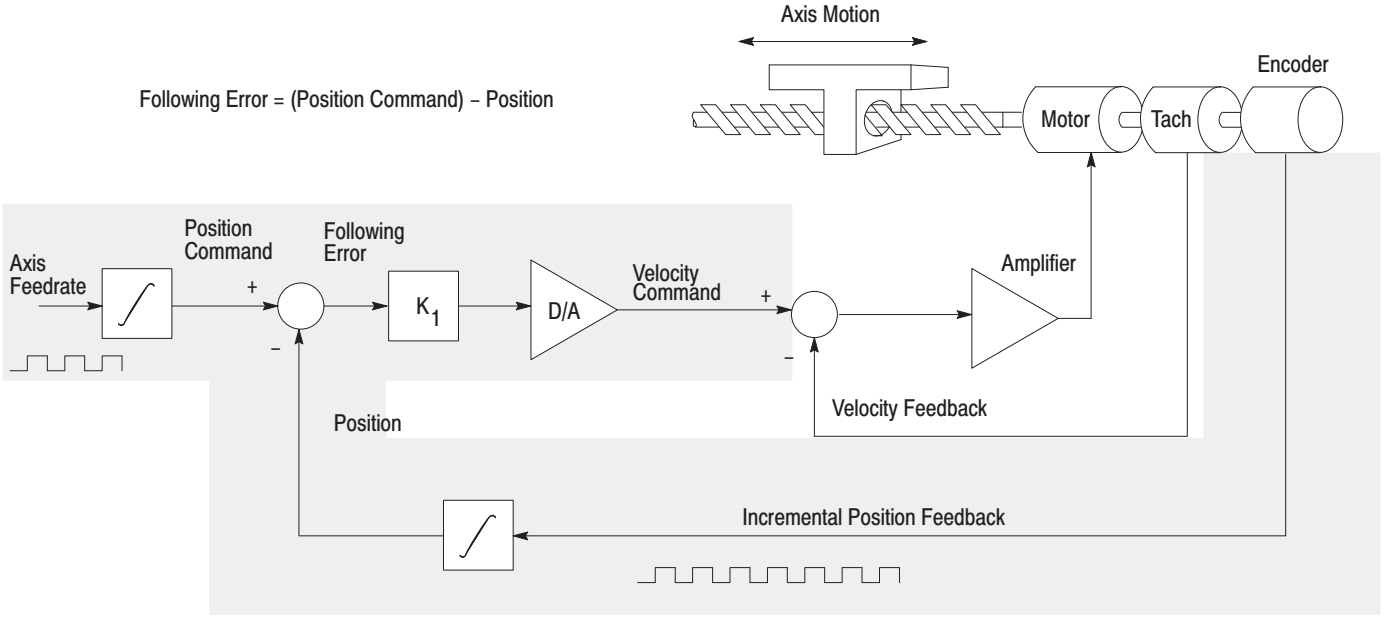


Whenever the velocity deviates from the commanded velocity, the velocity feedback signal adjusts the velocity error signal until the velocity matches the velocity command signal.

Positioning Loop

When we want to move the slide a specific distance, we can turn the motor on at a specific velocity for a specific length of time. However, this could produce imprecise positioning. To accurately control the position of the slide, we need a positioning loop (Figure 3.3).

Figure 3.3
Velocity Loop and Positioning Loop



The positioning loop includes a summing point, an amplifier, a D/A converter, and an incremental digital encoder to produce a position feedback signal. The axis feedrate is integrated in a register to produce the position command value. Incremental position feedback is integrated in a register to produce the actual position value. The position value is subtracted from the position command value. The difference is the following error, which is amplified and converted to an analog velocity command signal. This signal directs the axis to move in the right direction; the position value moves closer to the position command value.

The following error is a function of the axis velocity divided by the positioning-loop gain (K1). The following error is multiplied by the gain

to generate the velocity command. Gain is expressed in ipm/mil (where 1 mil - 0.001 in) or mmpm/mil (where 1 mil = 0.001 mm).

For example, with a velocity of 100 ipm and a gain of 1 ipm/mil, the following error is:

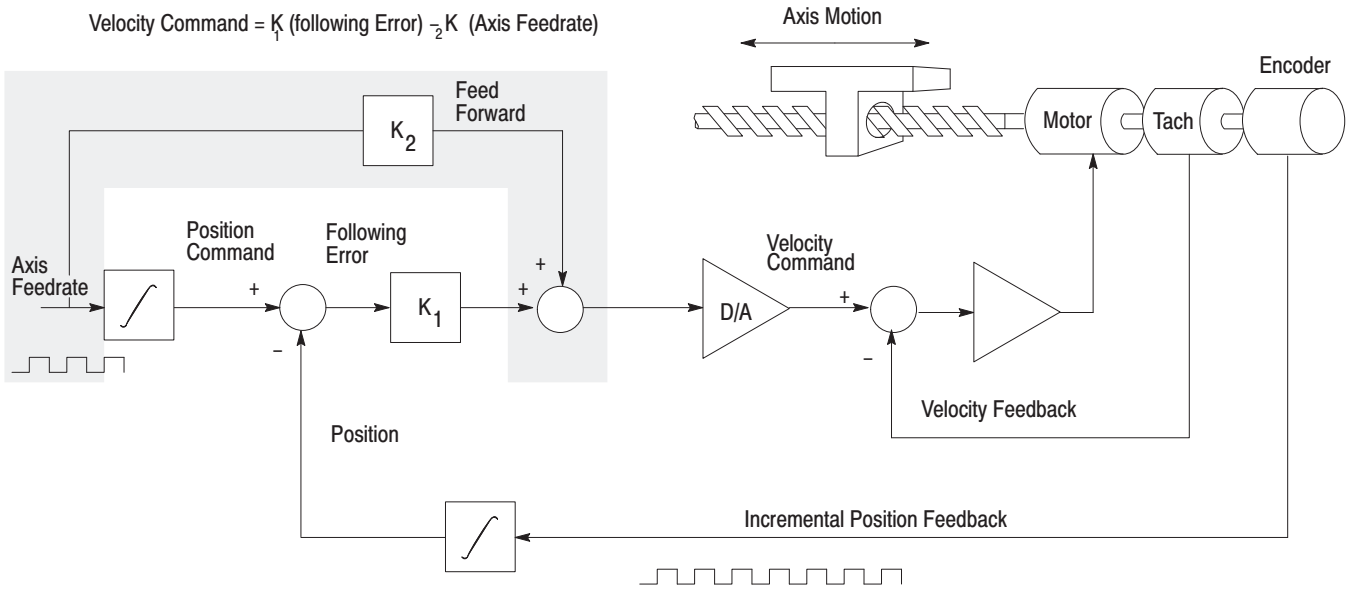
$$\text{following error} = \frac{\text{velocity}}{\text{gain}} = \frac{100 \text{ ipm}}{1 \text{ ipm/mil}} = 100 \text{ mil}$$

When you increase the gain, you decrease the following error and decrease the cycle time of the system. However, the gain that you can use is limited by the drive, the motor, and the machine; a gain that is too large causes instability.

Feed Forward

To decrease the following error without increasing the gain, we can add a feed forward component (Figure 3.4).

Figure 3.4
Velocity Loop, Positioning Loop, and Feed Forwarding



12002

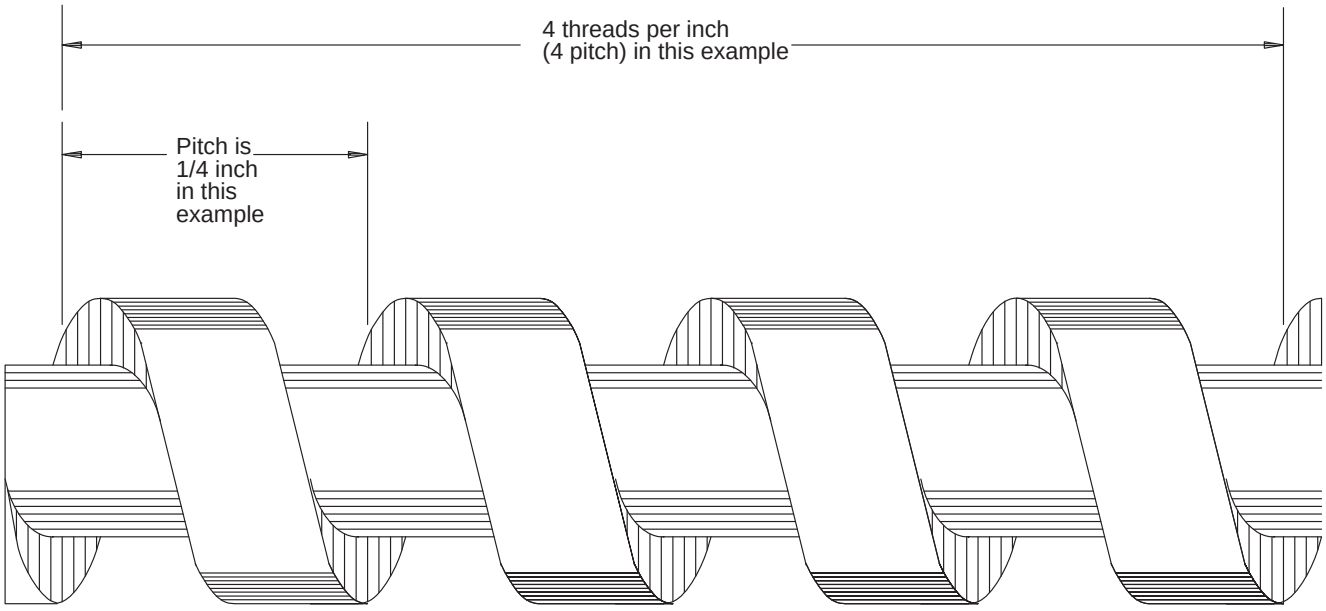
Feed forwarding requires an additional summing point and an amplifier. The axis feedrate is multiplied by the feed-forward gain (K2) to produce the feed-forward value. The feed-forward value is added to the following error multiplied by the gain to generate the velocity command.

Without feed forward, the axis will not begin to move until the axis feedrate builds up enough following error to generate a sufficiently large velocity command to overcome friction and inertia to move the axis. However, the feed-forward value could generate a velocity command to move the axis almost immediately. This immediate response keeps the actual position closer to the position command, thereby reducing the following error.

Leadscrew Pitch

Leadscrew pitch is the linear distance from one peak of the screw thread to the next. A leadscrew with a pitch of 1/4 inch is shown in Figure 3.5.

Figure 3.5
Leadscrew Example Showing Pitch



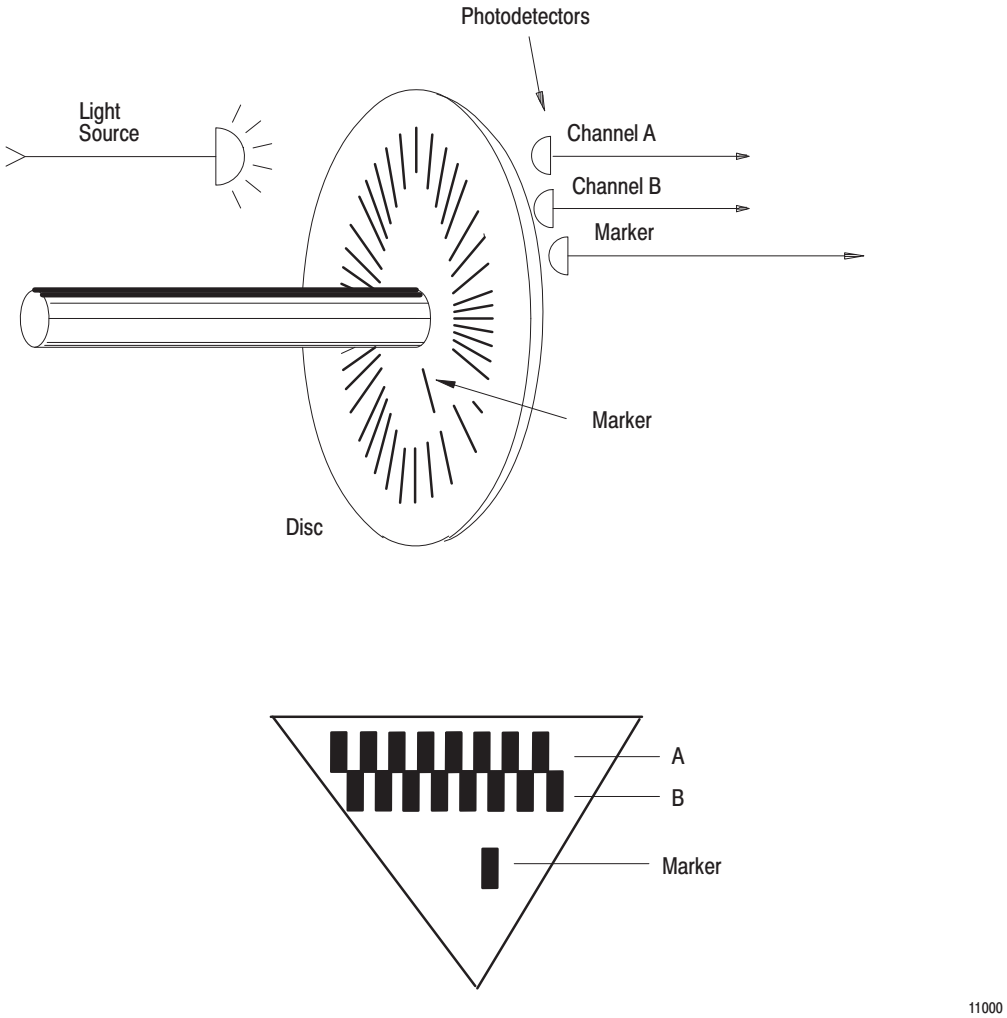
If the leadscrew has only one thread, the pitch is also equal to the lead, which is the distance the axis travels each revolution of the leadscrew. You can see from Figure 3.5 that the axis will travel 1/4 inch per revolution if the pitch is 1/4 inch. Since leadscrews normally have only one thread, and pitch is a more common term than lead, in this publication we use the term pitch to refer to the distance the axis travels for each revolution of the leadscrew.

Do not confuse leadscrew pitch with its inverse, which is the number of pitch (threads) per inch. In the example of Figure 3.5, the leadscrew has 4 pitch (threads) per inch. A leadscrew with a pitch of 1/4 inch is often described as being a 4-pitch (per inch) leadscrew.

Encoder Feedback

An incremental digital encoder provides feedback that indicates the magnitude and direction of any change of axis position. As shown in Figure 3.6, the encoder shaft is attached to a transparent disc marked with uniformly spaced lines. Strategically located photodiodes detect light. As the disc rotates, the lines break up the light reaching the photodiodes. As a result, the output (channel A, channel B, and marker) from each photodiode is a series of electrical pulses.

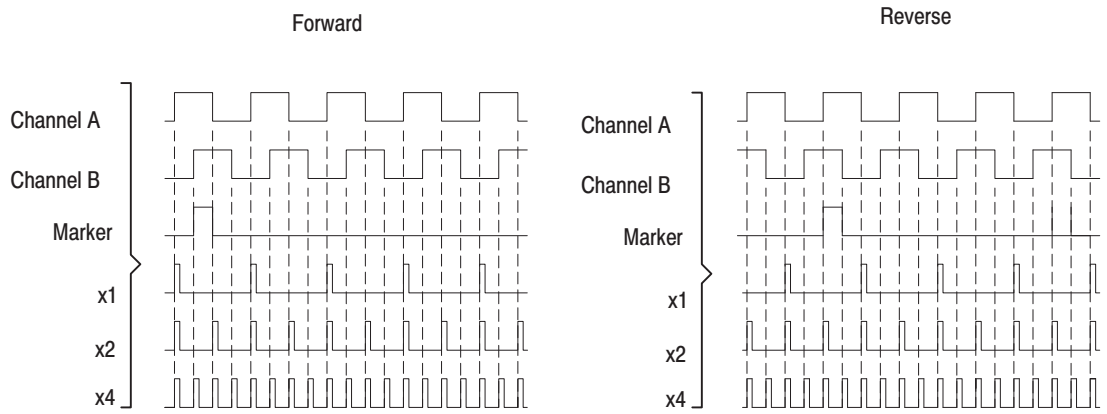
Figure 3.6
Incremental Encoder - Showing How Signals Are Generated



Channel Phase Relationship

The photodetectors are placed so that the channel A and channel B output signals are out of phase by 90º (Figure 3.7). The lead/lag relationship of these signals indicates the direction of axis motion. Also, the phase relationship of these signals allow the decoding circuit to count either 1, 2, or 4 feedback pulses for each line of the encoder (Figure 3.7). This provides flexibility in establishing feedback resolution.

Figure 3.7
Encoder Signals - Showing Phase Relationship



Note: For the servo positioning assembly, the encoder marker must be high when both channel A and channel B are high, or the marker is not recognized unless you set the marker logic jumper to the not-gated position.

11001

Feedback Resolution

The following discussion of feedback resolution assumes that you are using a leadscrew, and that the encoder is coupled directly to the leadscrew with no intermediate gearing. These assumptions apply to many applications. If your application differs, be sure to account for the differences.

Feedback resolution is the smallest axis movement the servo positioning system can detect. It is determined by:

- leadscrew pitch - axis displacement per revolution
- encoder lines - number of lines per revolution
- feedback multiplier - selected as x 1, x2, or x4

The following equation shows how these factors determine feedback resolution:

$$\text{feedback resolution} = \frac{\text{leadscrew pitch}}{(\text{encoder lines})\ (\text{feedback multiplier})}$$

You must select the leadscrew pitch, encoder lines, and feedback multiplier to provide desired feedback resolution and meet other requirements of your application.

The programming resolution of the servo positioning system is 0.0001 inch or 0.001 millimeter. If you select a feedback resolution coarser than that, round off your position commands so that the effective programming resolution is no finer than the feedback resolution you chose.

If you select a feedback resolution finer than the programming resolution, positioning can be smoother. However, the maximum axis speed is directly proportional to the feedback resolution. There is always a trade-off between feedback resolution and maximum axis speed. The maximum encoder input frequency for the servo positioning assembly is 250kHz. Therefore, to avoid a programming error, you must limit the axis speed to conform to this formula:

$$\text{programmed axis speed} < \frac{1.5 \times 10^7}{1.28 \times \text{feedback res} \times \text{feedback mult}}$$

The 1.28 factor allows for a 127% feedrate override value.

Each encoder line represents a fraction of a revolution of the leadscrew. For example, consider a 250 line encoder. Each line represents 1/250 of a revolution of the leadscrew.

Also, consider a 4-pitch (per inch) leadscrew for this example. The slide moves 1/4 inch for each revolution. With an x1 multiplier, each feedback increment represents 1/250 of 1/4 inch or 0.001 inch slide movement. This is the feedback resolution.

$$\text{feedback resolution} = \frac{0.25\ \text{in/rev}}{250\ \text{lines/rev} \times 1\ \text{increment/line}} = 0.001\ \text{in/increment}$$

Therefore, if we cause the leadscrew to move the slide 2 inches, we will get 2,000 feedback pulses.

Now, consider replacing the 250-line encoder with a 500-line encoder. By doubling the number of feedback pulses per revolution of the leadscrew, we improve the feedback resolution from 0.001 inch to 0.0005 inch.

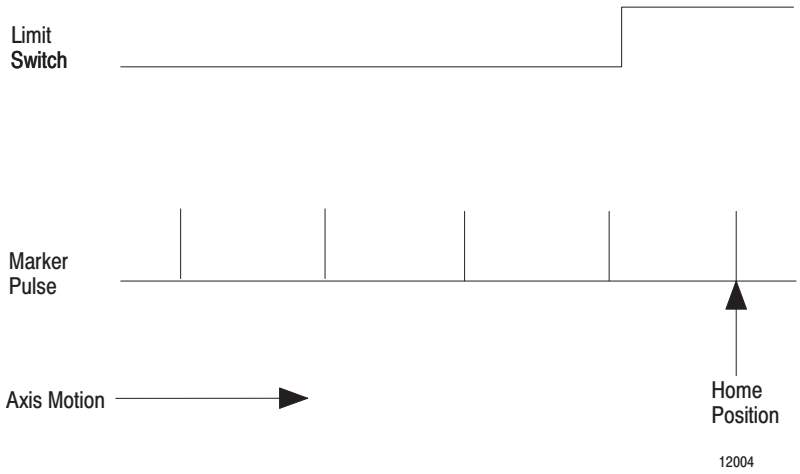
Another way to improve feedback resolution is to use a higher feedback multiplier. You can select a multiplier of x1, x2, or x4. For example, with the 4-pitch (per inch) leadscrew and the 250-line encoder, if you select an x2 multiplier you get the same feedback resolution improvement of from 0.001 inch to 0.0005 inch. With an x4 multiplier, you improve the feedback resolution to 0.00025 inch.

Marker

Besides the channel A and B output, an incremental encoder has a marker output (Figure 3.6 and Figure 3.7). The marker pulse occurs once every revolution. With a 4-pitch leadscrew, the marker pulse occurs at each 1/4 inch interval of slide travel.

We can use a market pulse to establish a home position somewhere along the slide travel. For example, we can place a limit switch near the end of the slide travel. The first market pulse after the limit switch is activated could then designate the home position (Figure 3.8).

Figure 3.8
Marker Pulse - Establishing a Home Position



Once we establish a home position, we can use it as an absolute reference point for all moves.

Summary

In this chapter we described concepts of closed-loop positioning. Now you are ready for concepts of position with an Allen-Bradley PC. This material is covered in chapter 4.

Chapter 4

Positioning With an Allen-Bradley Programmable Controller

Chapter Objectives

The previous chapter described concepts of closed-loop positioning. This chapter describes where the servo positioning assembly fits into a positioning system, and how the servo positioning assembly communicates with the PC processor.

Where the Servo Positioning Assembly Fits In

Figure 4.1 shows where the servo positioning assembly and a servo drive fit in the positioning system we described in the previous chapter. The servo drive contains the velocity loop summing point and amplifier. The servo positioning assembly contains the positioning loop summing point and the feed forward summing point. The servo positioning assembly sends the analog velocity command signal to the servo drive.

Figure 4.1
Where the Servo Positioning Assembly Fits in a Positioning System

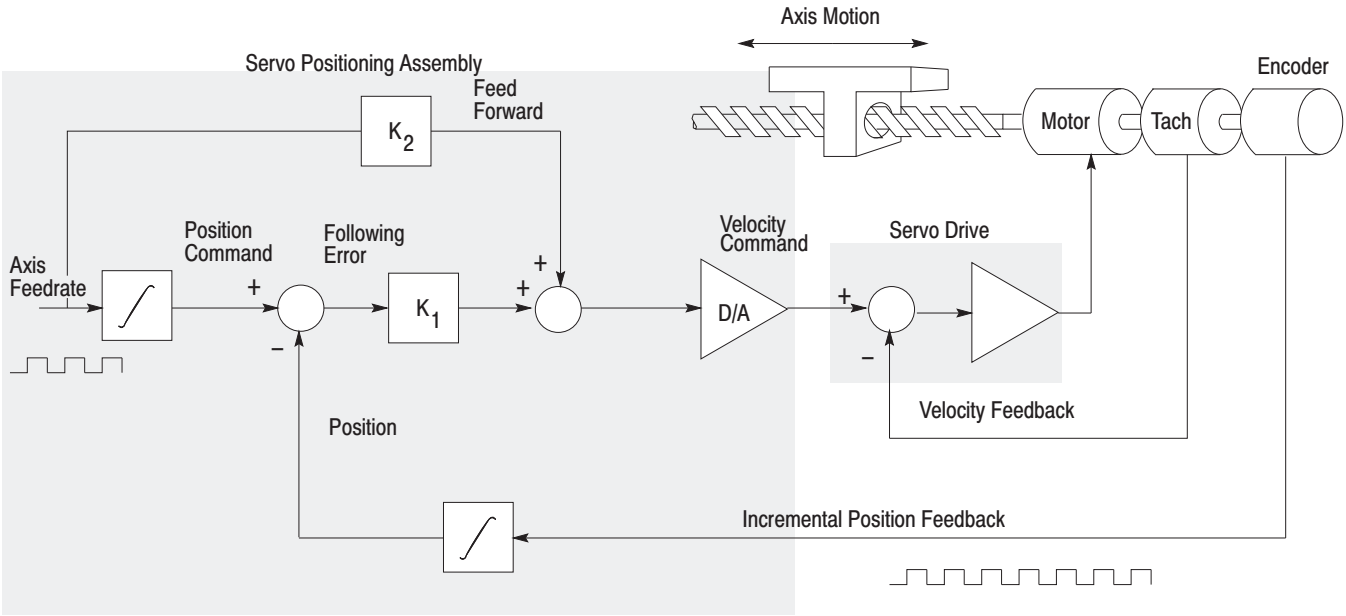
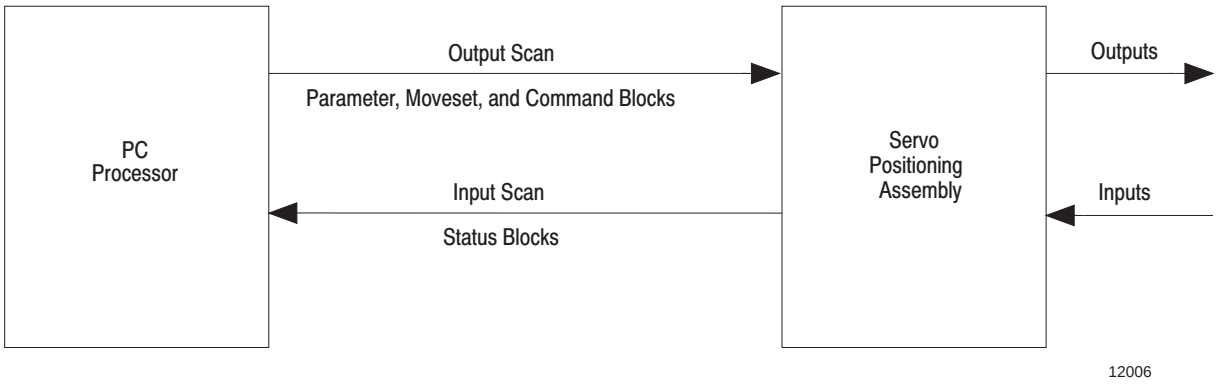


Figure 4.2 shows where the servo positioning assembly fits in a PC system. The PC processor constantly communicates with the servo

positioning assembly through the I/O scan. The PC processor acts on a block transfer read instruction to receive status blocks. Based on the status information received, the PC processor acts on a block transfer write instruction to send either parameter blocks, move blocks, or control blocks.

Figure 4.2
Where the Servo Positioning Assembly Fits in a PC System



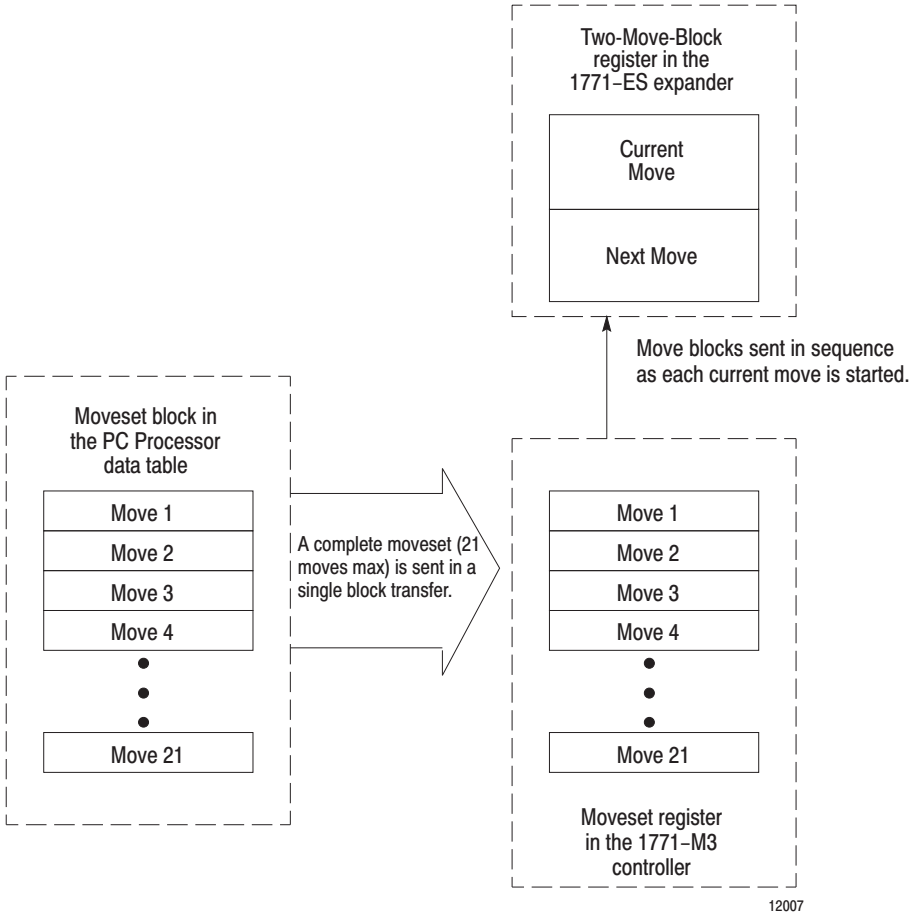
Independent of I/O Scan

Although the servo positioning assembly sends data to and receives data from the data table through the I/O scan, the positioning loop is closed on the 1771-ES expander (at the positioning loop summing point). This allows the 1771-ES expander to provide a servo sample period of 2.4ms, independent of I/O scan.

Move/Moveset

You must describe the axis motion you want in moveset blocks in the data table. You can enter a maximum of 21 separate move blocks in a moveset block (Figure 4.3).

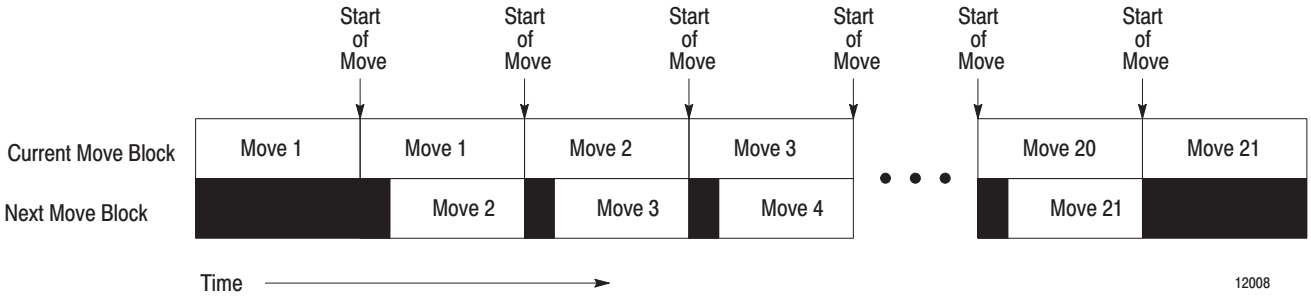
Figure 4.3
A Moveset Block is Sent to the 1771-M3 Controller That Sends the Move Blocks Sequentially to the 1771-ES Expander



The PC processor sends a complete moveset block to the 1771-M3 controller in a single block transfer. The 1771-M3 controller can hold a moveset block for each of the three possible axes.

The 1771-ES expander can hold two move blocks, the current move block available for execution and the next move block. After the current move is completed and the next move is to be executed, the next move block becomes the current move block (Figure 4.4).

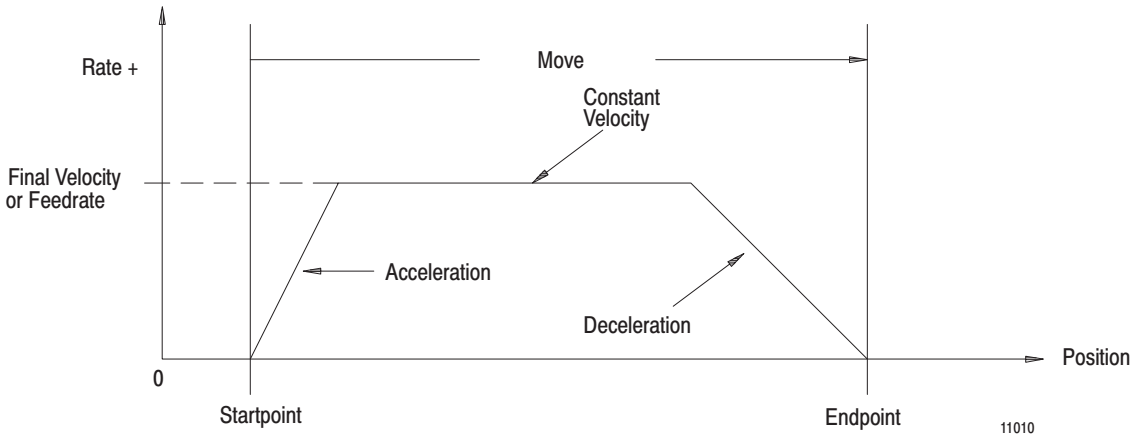
Figure 4.4
In the 1771-ES Expander, as Each Current Move is Completed, the Next Move Block is Ready to Take its Place



Initially, the 1771-M3 controller sends the first move block to the 1771-ES expander. Then, as each move is started the 1771-M3 controller sequentially sends each of the remaining move blocks to the 1771-ES expander.

A move block for a move to position defines motion of the axis from one position to another. Figure 4.5 shows the profile of an axis move. The horizontal axis in the figure represents axis position. The vertical axis represents axis velocity. Moves plotted above the position axis are in the positive direction (from left to right), moves plotted below the position axis are in the negative direction (right to left).

Figure 4.5
One-move Profile for an Axis



In the move shown in Figure 4.5, the axis:

- starts from a resting position
- accelerates to a final velocity

- moves at the final velocity some distance
- decelerates to zero velocity (at which time it has reached the programmed endpoint)

Move Values

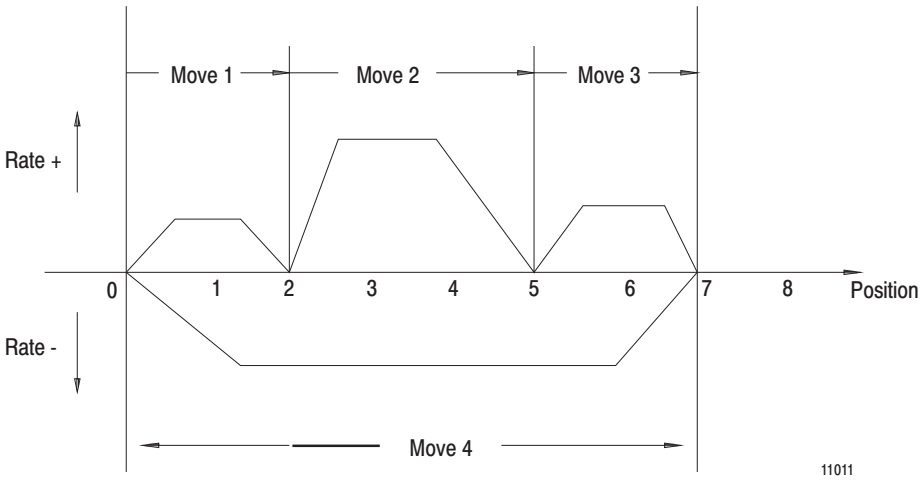
Each move block can specify several values. The servo positioning assembly executes the move based on these items you enter:

- endpoint
- acceleration
- final feedrate
- deceleration

When you select a deceleration value, the 1771-ES expander automatically calculates the point at which the deceleration must begin.

You can combine several single moves like that of Figure 4.5 to form a moveset. Figure 4.6 shows an example that consists of four moves. Move 1 starts at position coordinate 0 and ends at position coordinate 2. Move 2 continues axis motion to position coordinate 5. Move 3 continues to position coordinate 7. Move 4 then causes the axis to reverse direction and move back to position 0. The axis stops after it returns to its initial starting position. A drawing like that of Figure 4.6 is a moveset profile. You can use such profiles as an aid in programming axis motion.

Figure 4.6
Moveset Profile with All Single-step Moves



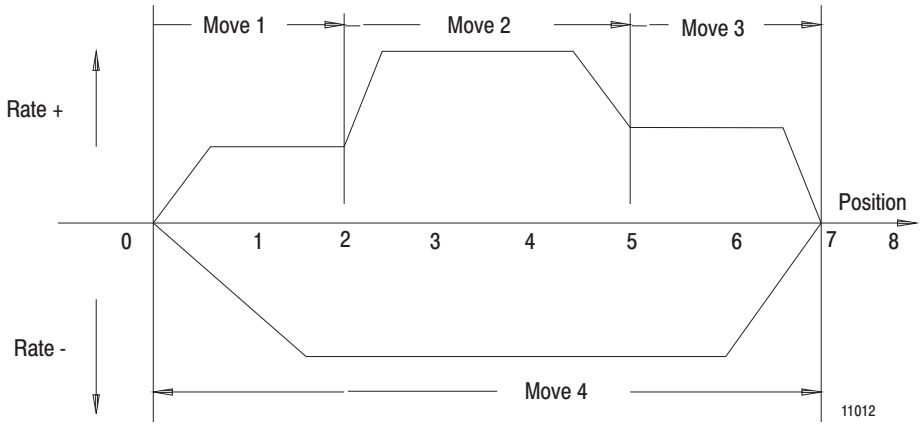
You can program multiple movesets for a given axis.

Move Selection

For each move, you have each of the following selections:

- **Absolute or incremental positioning** - In an absolute move, the endpoint value specifies a position coordinate relative to the current axis zero position. In an incremental move, the endpoint value specifies a position coordinate relative to the last programmed endpoint achieved by the axis.
- **Global or local values** - You enter a global final feedrate value and a global accel/decel rate value. These global rates apply to all moves except those for which you select to specify local rates. A local rate applies only to a single move.
- **Halt or run** - After completing a move for which you have selected halt, the 1771-ES expander will not execute the next move until it receives a begin or start command. After completing a move for which you have selected run, the 1771-ES expander will immediately execute the next move without waiting for a start command. With halt selected, the module executes a single-step move. With run selected, you can select moves to be either single-step moves or continuous moves.
- **Single-step or continuous** - When the 1771-ES expander executes a single-step move, it decelerates the axis to zero velocity at the programmed endpoint. When it executes a continuous move, it attempts to blend the move smoothly with the final feedrate of the next move (if the next move is in the same direction). The moves in Figure 4.6 are all programmed as single-step moves. Figure 4.7 shows the same moveset with all moves programmed as continuous. A moveset can contain a mix of single-step and continuous moves.

Figure 4.7
Moveset Profile with all Continuous Moves



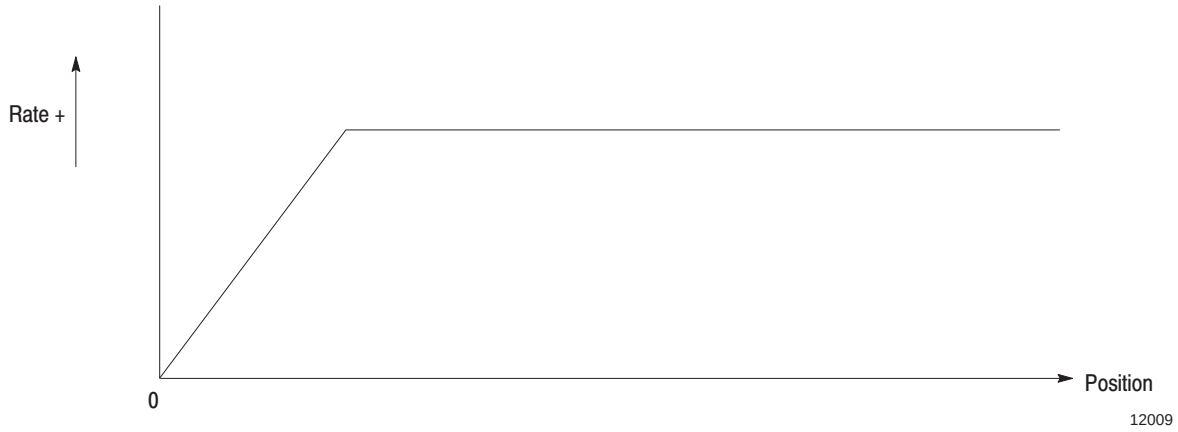
11012

Move Alternatives

In place of a move to position, in any move block you can select one of the following:

- **Dwell** - Instead of an endpoint and rates, you can program a time in seconds in the move block. When the 1771-ES expander executes a dwell move block, it stops axis motion for the programmed amount of time.
- **Preset to Position** - You can program an axis position preset value in the command block. When the 1771-ES expander executes a preset to position, it sets its axis position register to the programmed preset value. No axis motion occurs.
- **Move to Position with Offset** - The parameter block contains an offset value. When the 1771-ES expander executes a move to position with offset, it adds this offset value to an offset accumulator. For every move, it adds the value stored in the accumulator to the programmed endpoint then executes the move.
- **Constant Velocity** - This command clears the position register to zero before moving the axis to the position you specify. By repeatedly generating continuous constant velocity moves, you can cause uninterrupted motion, which could, for example, be applied to a conveyor.(Figure 4.8).

Figure 4.8
Moveset Profile for Constant Velocity Moves



In Position

For a continuous move with the next move in the same direction, the move is complete when the axis feed is done. The 1771-ES expander immediately begins the feedrate for the next move without waiting for the following error to close.

For any halt move, single-step move, or a continuous move with the next move in the opposite direction, the move is not complete until the axis is in position. The axis is in position when the following conditions are met:

- the axis feed is done
- following error has closed to within the in-position band

You establish the in-position band in the parameter block. The in-position band is the largest distance from the endpoint at which you will allow the axis to be considered in position.

Synchronizing Axes

In many applications it is important to synchronize the motion of two or more axes. In the following sections, we will tell you how to do this.

Halt Moves

For halt moves, axis synchronization is straightforward. When an axis is in position after a move, the next axis move will not begin until you send a start command.

You can monitor the in-position signal of each axis through the status block. When all axes are in position, you can send a start command to each axis through the command block.

Alternatively, you can monitor the in-position signal of each axis through the hardware done output terminal of the 1771-ES expander. When all axes are in position, you can send a start command to each axis through the hardware start input terminal of the 1771-ES expander.

Using the hardware start and done signals is faster than using block transfer for the status and command blocks. Furthermore, if the axis synchronization includes multiple servo positioning assemblies, precise synchronization cannot occur through block transfer because two block transfers cannot occur simultaneously.

Continuous Moves

For continuous moves with the next move in the same direction, axis synchronization requires precise programming of feedrates, acceleration rates, and deceleration rates. You must program the move blocks so that each axis takes the same amount of time for corresponding moves. Furthermore, you must plan the moves to be long enough to adhere to the following constraints:

- Each move must take longer than the time it takes to transfer a move block from the 1771-M3 controller to the 1771-ES expander. This time is a function of the number of axes as follows:

No. of Axes	Time
1	20ms
2	25ms
3	30ms

- If the number of moves requires additional moveset blocks, the last two moves of each preceding moveset block must not be too short. They must take a long enough time for the following moveset block to be transferred from the data table. (Refer to chapter 8 for details about block transfer timing.)

Run-Single-Step Moves

For run-single-step moves, axis synchronization is dependent upon the axis response on each move. The same is true for continuous moves with the next move in the opposite direction.

In both cases, the 1771-ES expander executes the next move automatically as soon as the current move is done, without waiting for a start signal. However, the time it takes for each move cannot be precisely calculated because the following error has to close before the move is done.

Auto Position Correction

The auto position correction feature may prevent an accumulation of position error caused by occasional noise on the channel A and B inputs. However, if the environment is excessively noisy, or if the cabling and shielding is not proper, this feature causes the axis to jump or jerk. This jump or jerk should indicate to you that a problem exists.

You enter the number of lines on the encoder and the feedback multiplier into the parameter block. From this, the 1771-ES expander knows how many feedback pulses it should receive each encoder revolution. The module also receives a marker pulse each revolution.

Each time the 1771-ES expander receives a marker pulse, it checks the value in the position register to see if it is an even multiple of the number of feedback pulses per revolution. If the value is off, the 1771-ES expander will automatically adjust it.

This feature corrects position errors caused by noise on the channel A and B encoder feedback signals. However, the function of this feature assumes a noise-free marker signal.

The marker signal does have some noise protection because the 1771-ES expander only accepts a marker signal when the channel A and B signals are high (unless you set the marker logic jumper to the not-gated position).

Specifying Axis Position

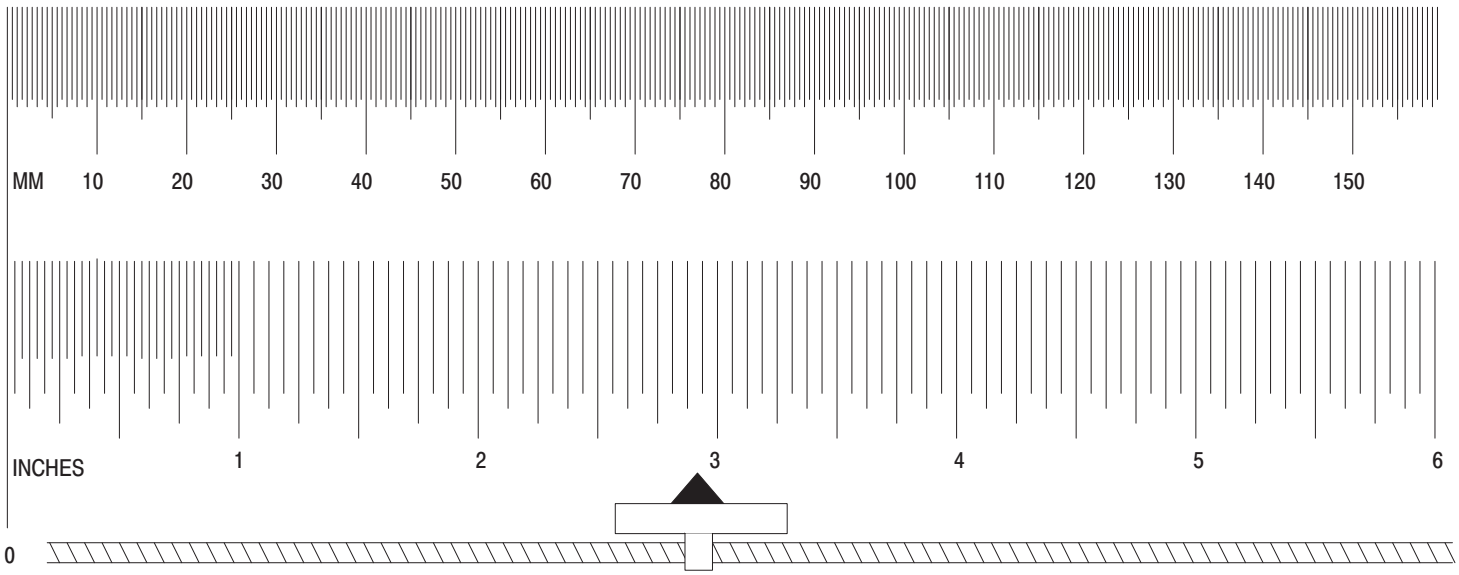
To command axis motion, you must be able to specify axis position by establishing an axis position scale, or coordinate system, for each axis.

Figure 4.9 shows an example of an axis and its position scale. Any axis position within the range of travel can be identified by a number. For the servo positioning assembly, the axis position scale can be either in inches or millimeters.

The position scale is an internal scale used by the servo positioning assembly to identify axis position. It is not printed on the axis slide. You can shift the axis position scale by entering (through the command block) any of the following commands:

- search home
- preset
- initialize home

Figure 4.9
Axis Position Scale

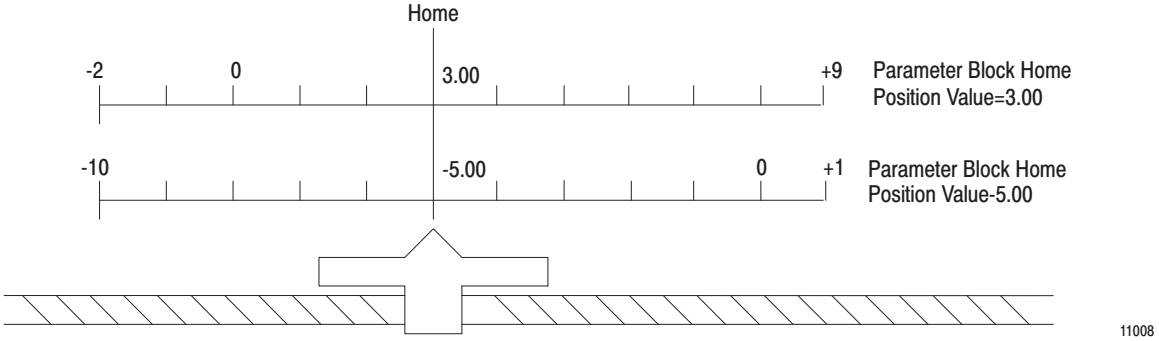


Search Home

Because the position feedback is incremental rather than absolute, the servo positioning assembly does not know the axis position when it first receives power. You must command a search home (through the command block) each time after powering up. In the search home operation, the axis moves until the servo positioning assembly detects the first encoder marker beyond the user-installed home limit switch. The

axis stops on the marker. The servo positioning assembly then sets it position register to the home position value you specify in the parameter block. This initializes the axis position scale. Figure 4.10 shows how the home position value you specify in the parameter block can affect the axis position scale. This figure compares the scales for an axis after search home operations with different home position values form the parameter block representing the same physical position.

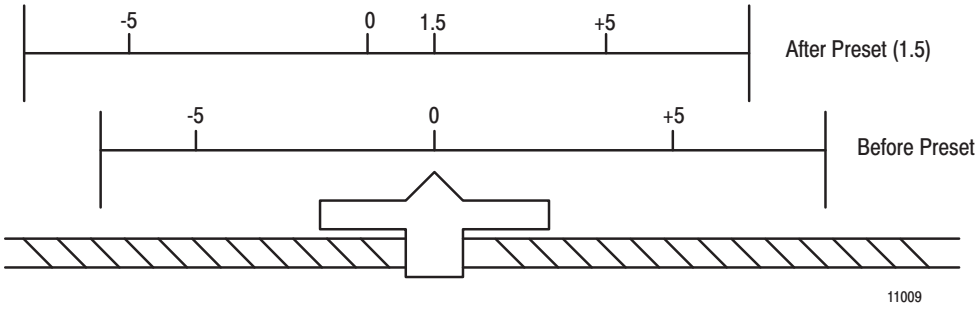
Figure 4.10
Axis Position Scales for 2 Home Position Values



Preset

Through a command block, you can command the servo positioning assembly to preset a specified value into its position register. When the servo positioning assembly executes a preset command, it sets its position register to the specified value without causing axis motion. This action effectively shifts the axis position scale. Figure 4.11 shows an axis position scale before and after a preset operation.

Figure 4.11
Axis Position Scale before and after Preset



Initialize Home

Through a command block you can generate an initialize home command. The initialize home operation assigns the home position value (which you specify in the parameter block) to the current axis position. Its effect is the same as that of the preset operation, except that the new position value is the home position value.

Summary

Now that you have been familiarized with the general concepts of how the servo positioning assembly functions in a closed-loop positioning system and in a PC system, you are ready for specific details of the servo positioning assembly in chapter 5.

Chapter 5

Hardware Description

Chapter Objectives

The previous chapter described how the servo positioning assembly fits into a positioning system as part of a programmable controller. This chapter describes specific hardware of the servo positioning assembly and lists its specifications. This chapter also describes other hardware items you need for a positioning system.

Indicators

There are three indicators on the 1771-M3 controller. With the PC processor operating in the run mode, the indicators have the following functions:

- Processor Communication Fault - This red indicator turns on when the module detects a fault in the communication between it and the PC processor. The I/O adapter module or PC processor will not detect this as a fault.
- Expander Communication Fault - This red indicator turns on when the module detects a fault in the communication between it and a 1771-ES expander.
- Active - This green indicator is normally on. It turns off when a hardware fault is detected on a 1771-ES expander. it blinks if you have not properly configured the modules.

There are six indicators on the 1771-ES expander. With the PC processor operating in the run mode, the indicators have the following functions:

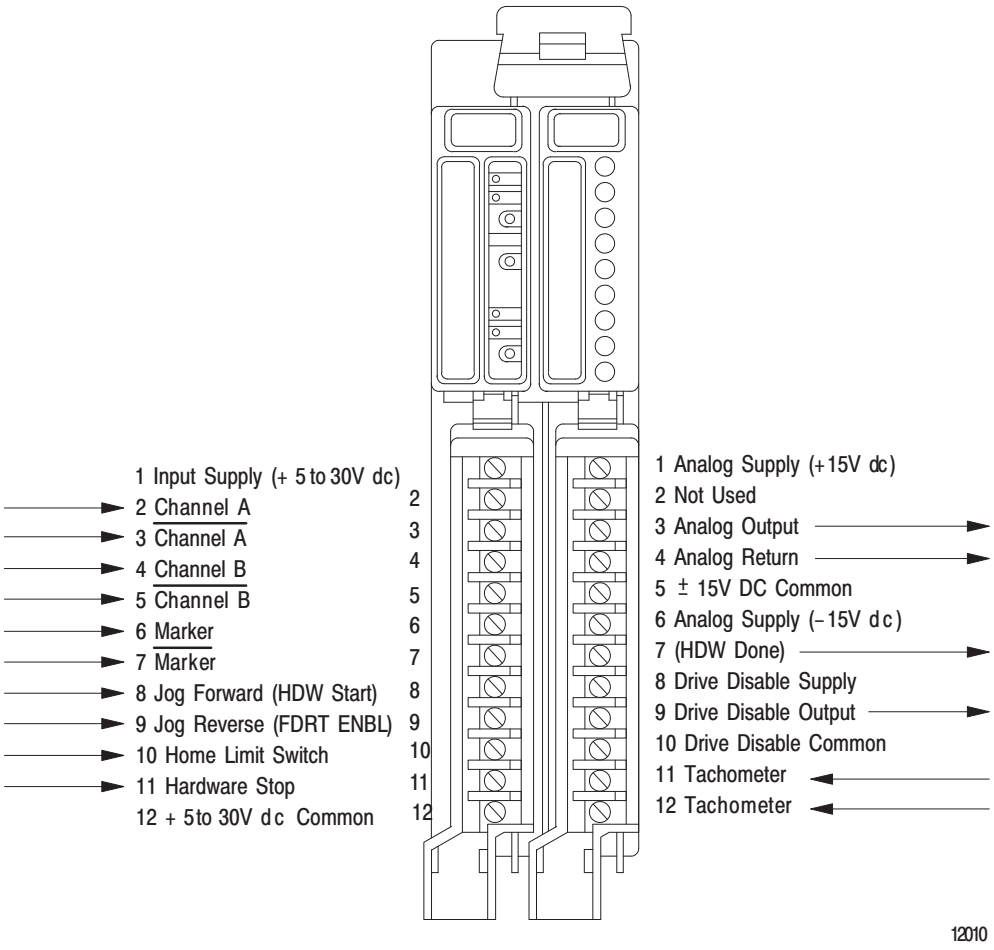
- Module Active - This green indicator is on when the module is operating normally.
- Marker - This green indicator is on when the channel A, channel B, and marker signals are true simultaneously.
- Home - This green indicator is on when the axis is in the home position.
- Tach Calibrate - This green indicator is used in setting the adjustments for loss of feedback detection.
- Hardware Stop - This red indicator goes on when the hardware stop input opens. It stays on until the input closes and the servo expander module is reset.
- Diagnostic - This red indicator goes on when a fault is detected at the servo expander module.

These indicators are useful troubleshooting aids, described fully in chapter 9.

Inputs/Outputs

The 1771-M3 controller requires no connections. You will make all wiring connections to the 1771-ES expander. Figure 5.1 shows the terminals on the 1771-ES expander. These terminals provide the connection points for all the inputs and outputs of the servo positioning assembly. Limit the cable length to 50 feet for all connections.

Figure 5.1
Terminals On the 1771-ES Expander Showing Input and Output Signals



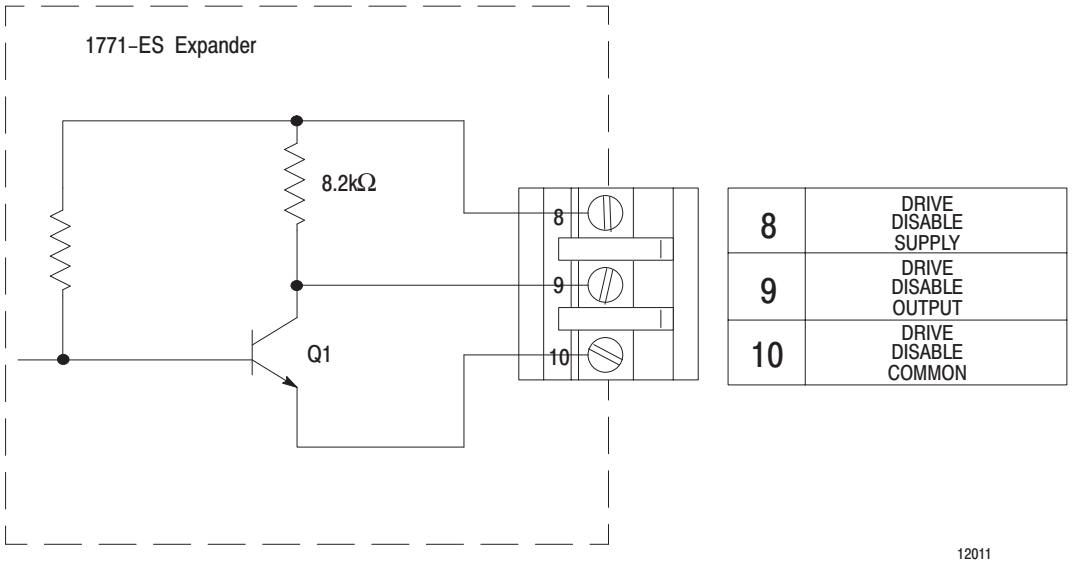
Outputs to Servo Drive

Terminals 3 and 4 on the right wiring arm provide connection points for the velocity command signal to the serve drive. This analog output is a +10V dc differential signal.

Terminals 8, 9, and 10 on the right wiring arm provide connection points for a drive disable signal (Figure 5.2). In chapter 6 we will show you how to connect this output to either source or sink 100mA maximum to enable the drive. The module normally provides current thru this transistor to enable the drive. However, the module will turn off the current to disable the drive if:

- the hardware stop input goes high
- a command block commands an immediate stop
- a firm ware or hardware watchdog timers times out
- the 1771-ES expander detects excess following error, a loss of feedback, or a power supply loss

Figure 5.2
Schematic Diagram of the Drive-disable Output Circuit



The 1772-ES expander is compatible with a wide variety of servo drives, including Allen-Bradley Bulletin dc Servo Controllers (refer to publication 1388 -5.0). Allen Bradley also offers Bulletin 1326 dc servo Motors to match the Bulletin 1388 dc Servo Controllers.

Tachometer Input

Terminals 11 and 12 on the right wiring arm provide connection points for the velocity feedback signal from the tachometer. Although the velocity loop is closed on the servo drive, the 1771-ES expander uses the velocity feedback signal to compare to the position feedback signal from the encoder. If the module detects an imbalance between these signals, it disables the servo drive and sends a loss of feedback signal through the status block.

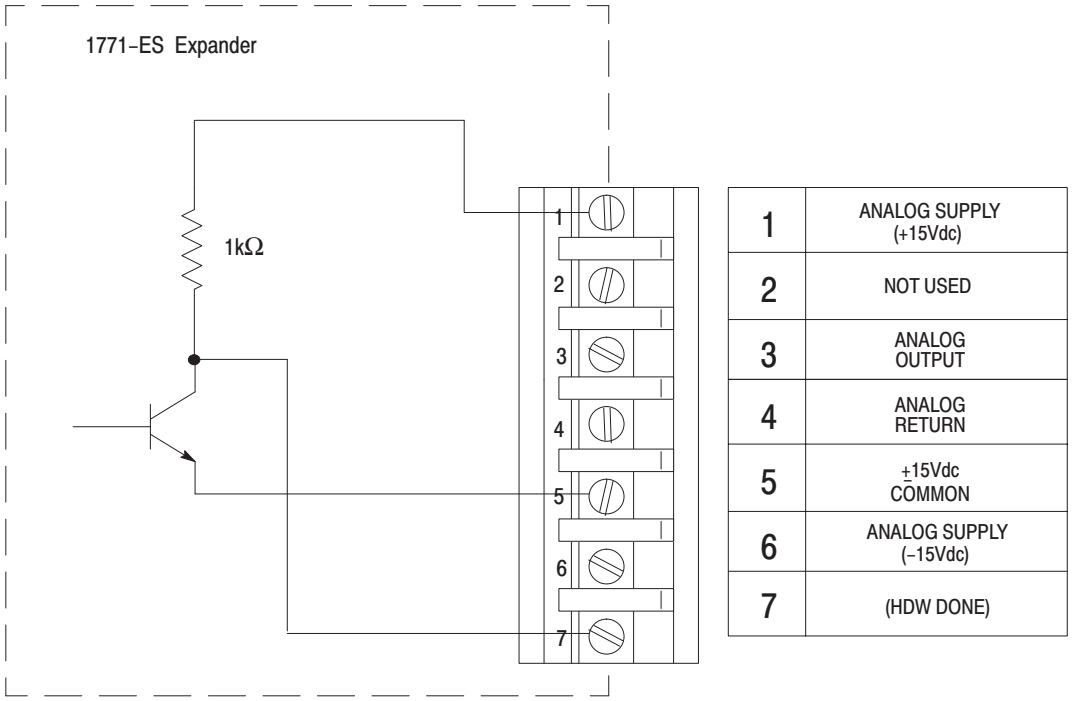
The 1771-ES expander accepts a full scale tachometer signal of 3V to 50V dc. If the full scale tachometer signal is greater than 50V dc, you must reduce it through a voltage divider on the servo drive before connecting it to the module.

CAUTION: Do not connect a signal greater than 50V dc across these terminals. A signal greater than 50V dc could damage the 1771-ES expander.

Hardware Done Output

Terminal 7 on the right wiring arm provides a connection point for a hardware done output signal (Figure 5.3).

Figure 5.3
Schematic Diagram of the Hardware-done Output Cirucit



The output transistor, normally on, provides a 15mA (maximum) sink. When the axis feed is done and the axis is in position, the transistor is off and the circuit provides +15V dc through a 1k resistor. This provides you with a hardware done signal that is high-true.

In chapter 6, we will show you how to connect the hardware done signal to a dc (12-24V) Input Module (cat. no. 1771-IB) for axis synchronization of halt moves.

Discrete Inputs

Terminals 8, 9, 10, and 11 on the left wiring arm provide connection points for discrete input signals. The module accepts a discrete input signal as being high when it reaches 40% of the input power supply voltage. The module accepts a discrete input signal as being low when it reaches 20% of the input power supply voltage.

Each discrete input has an internal pull-up resistor. In chapter 6, we will show you how to select an internal pull-up resistor of 1.2k or 11.2k. You select each input individually through a switch setting.

For a high signal, the input device you connect to a discrete input does not have to source current. For a low signal, the input device you connect to a discrete input has to sink current through the pull-up resistor.

Hardware Start

In the auto mode, the module accepts a high-to-low transition at terminal 8 of the left wiring arm as a low-true hardware start input signal.

After completing a halt move, the 1771-ES expander will not execute the next move until it receives a start command. The start command could come through block transfer of a control block or through the hardware start signal.

Feedrate Override Enable

In the auto mode, the module accepts a high-to-low transition at terminal 9 of the left wiring arm as a **low-true** feedrate override enable signal.

After setting a feedrate override value for the axis through the command block and enabling external synchronization of feedrate override through the parameter block, you can enable the feedrate override through this input. Do this by setting bit 16 of word 17 in the parameter block ON (Axis 1). (Set bit 16 of words 36 and 55 for axis 2 and 3, respectively.). This allows you to activate a preloaded feedrate override value to change speed on several axes at the same instant.

Jog Forward

In the manual mode, the module accepts the signal at terminal 8 of the left wiring arm as a **low-true** jog forward signal. When the module receives this signal, it moves the axis in the positive direction at the rate established through block transfer.

Jog Reverse

In the manual mode, the module accepts the signal at terminal 9 of the left wiring arm as a **low-true** jog reverse signal. When the module receives this signal, it moves the axis in the negative direction at the rate established through block transfer.

Home

The module accepts the signal at terminal 10 of the left wiring arm as a **low-true** home signal. The module considers the first marker pulse after the home signal as the home position.

Hardware Stop

The module accepts the signal at terminal 11 of the left wiring arm as a **high-true** hardware stop signal. Unless this input is pulled low, the module holds the velocity command output signal at zero and disables the servo drive by turning off the drive disable circuit.

Encoder Inputs

Terminals 2, 3, 4, 5, 6, and 7 on the left wiring arm provide connection points for input signals from the encoder. Through jumpers on the module, you can select each channel individually for either single-ended or differential, and for either high-true of low-true input signals.

If you use a single-ended encoder, limit the input pulse rate to 20k Hz. If you use a differential encoder, limit the input pulse rate to 250k Hz.

The 1771-ES expander is compatible with Allen-Bradley Incremental Differential Line Driver Encoders (cat. no. 845N-SJDN4-C) and with other encoders having current-sinking (5-30V dc) line-driver outputs, totem-pole (TTL) outputs, or open-collector outputs.

External Power Supplies

You must provide at least two external dc power supplies to provide power for the input and output circuits.

Input Supply

You must connect a 5-30V dc power supply between terminals 1 and 12 of the left wiring arm. This provides power for the input circuits. The input circuits require 500mA (maximum) at 30V. You can use the same power supply to power the encoder if the power supply has enough additional current capacity for the encoder.

Drive Disable Supply

Unless the servo drive provides its own dc voltage source for this circuit, you'll need a 5 - 30V dc power supply to provide 100mA (maximum) for the drive disable circuit. How you connect this power supply depends on whether the servo drive requires a current source or a current sink to enable it.

Analog Supply

A separate +15V dc supply is needed to provide 200mA (maximum) for the digital/analog converter (DAC) to generate the analog output signal and for the hardware done output circuit.

Compatible Processors

The servo positioning assembly can be used with PC processors that have block transfer capability and adequate data table size to contain the data blocks you need for your application. Compatible PC processors include:

- Mini-PLC-2/05 (cat. no. 1772-LS,-LSP)
- Mini-PLC-2/15 (cat. no. 1772-LV)
- PLC-2/20 (cat. no. 1772-LP2)
- PLC-2/30 (cat. no. 1772-LP3)
- PLC-3 (cat. no 1775-L1,-L2)

Fault Responses

The servo positioning assembly provides a means for detecting and responding to faults in your servo positioning system.

Since the servo positioning assembly is part of a PC system, diagnostic information about fault conditions detected by the servo positioning assembly can be block transferred to the PC processor.

At the PC processor, you can use the ladder diagram program to respond to diagnostic information about fault conditions in any way you feel is appropriate for your application. This may include turning off machinery, turning on alarms, or generating report printouts. Furthermore, with an Allen-Bradley Data Highway network, you can send this diagnostic information to a computer or other Allen-Bradley PC processors.

The servo positioning assembly provides specific fault responses if certain critical connections are broken.

Loss of Feedback

The 1771-ES expander continuously monitors the tachometer and encoder feedback. If it senses an imbalance between these signals, it holds the velocity command output signal at zero and disables the servo drive through the drive disable circuit. Therefore, if the cable from either the encoder or the tachometer breaks, the 1771-ES expander will disable the servo drive.

Hardware Stop

You must connect a set of normally open contacts of your master control relay between the hardware stop input terminal and the input power supply common terminal. Normally, the master control relay would be energized, pulling the hardware stop input low. This allows the module to enable the servo drive.

However, if the master control relay de-energizes for any reason (such as extreme overtravel limit or emergency stop), the hardware stop input goes high. This forces the module to hold the velocity command output signal at zero and disable the servo drive by turning off the drive disable circuit. Therefore, if a connection in the hardware stop circuit breaks, the 1771-ES expander will disable the servo drive.

Loss of Power

The 1771-ES expander holds the velocity command output signal at zero and disables the servo drive by turning off the drive disable circuit if it is unable to sense the specified voltage as the following power-supply terminals:

- positive (+) terminal for the input power supply
- common (-) terminal for the input power supply
- positive (+) terminal for the analog power supply
- negative (-) terminal for the analog power supply

Therefore, if one of these power supplies connected to the 1771-ES expander terminal fails or if one of these connections from these power supply breaks, the 1771-ES expander will disable the servo drive.

The drive disable circuit normally provides current to a sensing circuit on the servo drive to enable it. However, if the 1771-ES expander detects a fault, it cuts off the current in the drive disable circuit, thereby disabling the servo drive. Therefore, if a connection in the drive disable circuit breaks, this disconnection will disable the servo drive.

Auto Position Correction

Each time the 1771-ES expander receives a marker pulse, it checks the value in the position register to see if it is an even multiple of the number of feedback increments per revolution. If the value is off, the 1771-ES expander will automatically adjust it to the closest even multiple.

This auto position correction feature corrects position errors caused by noise on the channel A and B encoder feedback signals. However, the function of this feature assumes a noise-free marker signal. Although this feature may be able to prevent an accumulation of position error caused by occasional noise on the channel A and B inputs, it cannot maintain position accuracy if the environment is excessively noisy or if the cabling and shielding is not proper.

If the environment is excessively noisy or if the cabling and shielding is not proper, this feature will cause the axis to jump or jerk. This jump or jerk indicates a problem.

Note that when the module detects a position error, it does not necessarily disable the servo drive.

Because this feature adjusts the position register to the closest even multiple of the number of feedback increments per revolution, it is essential that the axis move less than half an encoder revolution per servo sample period (2.4ms). Therefore, to avoid a programming error, you must limit the axis speed to conform to this formula:

$$\text{programmed axis speed} < \frac{12{,}500}{1.28} \times FR \times FM \times EL$$

Where:

FR = feedback resolution
FM = feedback multiplier (1, 2, or 4)
EL = encoder lines per revolution

Specifications

Here is a list of specifications for the servo positioning assembly.

Servo Output Voltage

- ±10V dc maximum (isolated)

D/A Converter (DAC)

- Signed 12 bit resolution

Encoder Input

- High: 1.6V
- Low: 1.0V sinking lmA

Encoder Input Rate

- Differential: 250k Hz maximum
- Single-ended: 20k Hz maximum
- Jumper selection of differential or single ended input

Encoder Multiplier

- x1, x2, or x 4,programmable

Tachometer Input (For loss-of-feedback detection)

- Full scale voltage: 3V dc minimum, 50V dc maximum
- Input impedance: 20k ohmss

Discrete Inputs

- Resistance to high side of supply 11.2k ohms or 1.2k ohms, switch selectable for each input
- For a low, required sink current with 1.2k ohms resistance: 4mA @ 5V, 24mA @ 30V
- For a low, required sink current with 11.2k ohms resistance: 0.4mA @ 5V, 2.7mA @ 30V
- High: 40% of + dc supply voltage
- low: 20% of + dc supply voltage

Hardware Done Output

- On: +15V source thru 1k ohms resistance
- Off: 15mA sink

Drive Disable Output

- Current: 100mA maximum, source or sink
- Voltage: 30V dc maximum to 5V dc minimum

Backplane Current

- 1771-M3 controller: 1.75A
- 1771-ES expander: 1.70A

External Power Supply Requirements

- External supply for inputs, +4.75 dc minimum, +30V dc maximum, 500mA maximum
- External supply for DAC and hardware done output, +15V dc, 200mA maximum
- External supply for drive disable output, +4.75V dc minimum, +30V dc maximum, 100mA maximum

Maximum Programmable Position

- ±999.9999 inches (resolution 0.0001 inch)
- ±19999.999 millimeters (resolution 0.001 mm)

Programmable Speed

- 0.0001-9990.0000 ipm (resolution 0.0001 ipm)
- 0.001-199900.000 mmpm (resolution 0.001 mmpm)

Accel/Decel

- 9999 ipm/s maximum (resolution 1 ipm/s)
- 99.99 mpm/s maximum (resolution 0.01 mpm/s)

Initial Servo Gain (Programmable)

- 0.01-9.99 ipm/mil following error (1 mil = .001 inch)
- 0.01-9.99 mmpm/mil following error (1 mil x .001 mm)

Servo Sample Period

- 2.4ms

Environmental Conditions

- Operational Temperature: 0º to 60ºC (32º to 140ºF)
- Storage Temperature: -40º to 85ºC (-40º to 185ºF)
- Relative Humidity: 5% to 95% (without condensation)

Keying

- Servo controller slot: between 2 and 4, 8 and 10
- Left servo expander slot: between 2 and 4, 14 and 16
- Right servo expander slot: between 4 and 6, 32 and 34

Summary

Now that you have read about the function of each input and each output, you are ready to install the servo positioning assembly. Chapter 6 gives you this information.

Chapter 6

Installing the Assembly

Chapter Objectives

The previous chapter described the hardware of the servo positioning assembly. This chapter tells you how to install the servo positioning assembly. As you install it, you will make hardware selections to direct its operation to fit your application requirements.

Configuring the Modules

The first step of installing a servo positioning assembly is to plan how to configure modules in the I/O chassis.

Planning Module Combinations

You can install one 1771-M3 controller in an I/O chassis together with either one, two, or three 1771-ES expanders. However, the I/O chassis must not contain any other module combination of a master (such as an analog module) and its slave (expander).

A master must communicate with its slaves through the backplane. Two masters trying to communicate through the backplane interferes with each other.

If you have an illegal combination of 1771-ES expanders or a second master/slave combination in the I/O chassis, the **active** indicator on the 1771-M3 controller blinks. An illegal combination of 1771-ES expanders would be:

- the number of 1771-ES expanders not matching the number of axes in the parameter block
- an axis 2 with no axis 1
- an axis 3 with no axis 2
- two axes with the same number

Always use the same series level of 1771-M3 controller and 1771-ES expander. You cannot use a series A 1771-M3 controller with a series B 1771-ES expander. Likewise, you cannot use a series B 1771-M3 controller with a series A 1771-ES expander.

Avoiding Backplane Power Supply Overload

For each module you plan to install in the I/O chassis, add up it current load on the backplane power supply. Be sure that this total current is not so large as to overload the backplane power supply.

The backplane power supply current load of the servo positioning assembly is:

1771-M3 controller	1771-ES expanders	Total Current
1	1	3.45A
1	2	5.15A
1	3	6.85A

Note that if you add the total current draw of one 1771-M3 controller, three 1771-ES expanders, and either an I/O adapter or mini-processor module, the total would exceed 8A. In that case you could not use a 1771-P1 or 1771-P2 power supply because they are rated at 6.5A.

If the total current exceeds 6.5A, you can use Power-Supply Modules (cat. no. 1771-P3, -P4, -P5) to provide 8A, 11A or 16A. The following table lists the number of axes you can control with a servo positioning system in a 1771-A4 I/O chassis, based on power requirements and compatibility of other components used with the 1771-A4 I/O chassis.

Power Supply Cat. No.	I/O Adapter or Mini-Processor Module Cat. No.				
	1771-AL	1771-AS	1772-LS	1772-LSP	1771-LV
1771-P1					1 Axis
1771-P2	2 Axes	2 Axes			
1771-P3	2 Axes			1 Axis	
1771-P4	3 Axes		3 Axes	3 Axes	
1771-P4 plus 1771-P3 or a second 1771-P4	3 Axes		3 Axes		

Planning Module Location

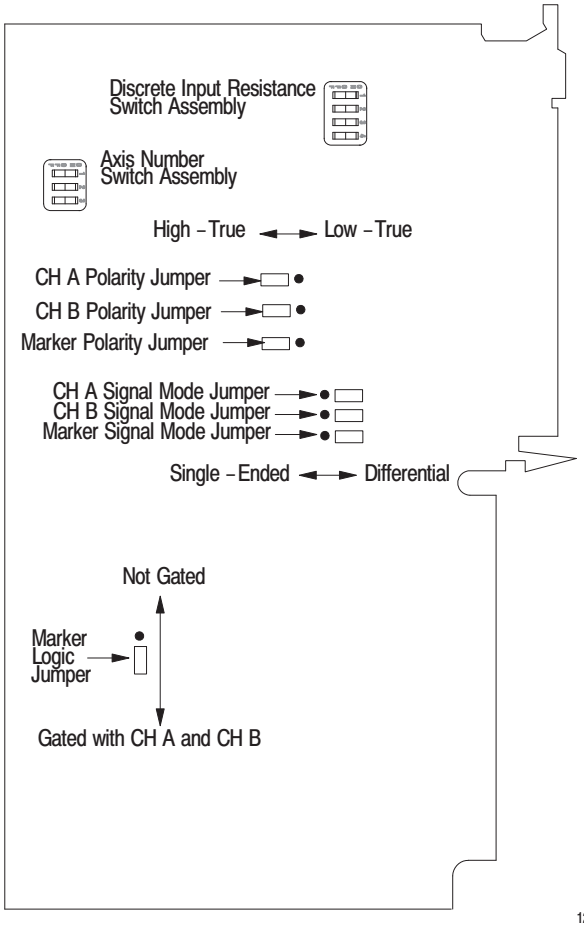
The 1771-M3 controller requires one I/O chassis slot. You can install it in any I/O in the I/O chassis. The 1771-M3 controller uses both the output image table byte and the input image table byte that correspond to its location address.

The 1771-ES expander requires two slots. Install it in a pair of slots that make up an I/O module group.

Setting Switches and Jumpers

Through switches and jumpers on the 1771-ES expander, you can select various aspects of the module's operation. To access these switches and jumpers, lay the 1771-ES expander on its right side and remove the left cover. Locate the switches and jumpers through Figure 6.1.

Figure 6.1
1771-ES Expander Switches and Jumpers



This publication shows and describes switches as being on or off. Printed on the actual switch assemblies are the words ON and OFF or the word OPEN. OPEN corresponds to OFF.

Use a blunt-pointed instrument such as a ball-point pen to set these switches. **Never** use a pencil; graphite could jam the switch.

Figure 6.2 shows details of a jumper connecting two pins. Each jumper connects two of a set of three pins. To change a jumper setting, follow these steps:

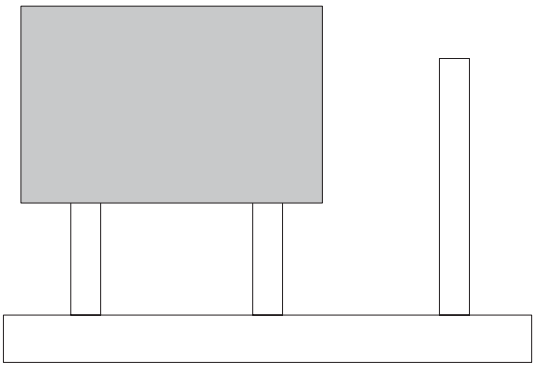
1. Pull the jumper straight up.

2. Position the jumper over the pins you want to connect.

3. Push the jumper straight down.

If you position the jumper correctly, it slides down over the pins easily.

Figure 6.2
Jumper in the Left Position

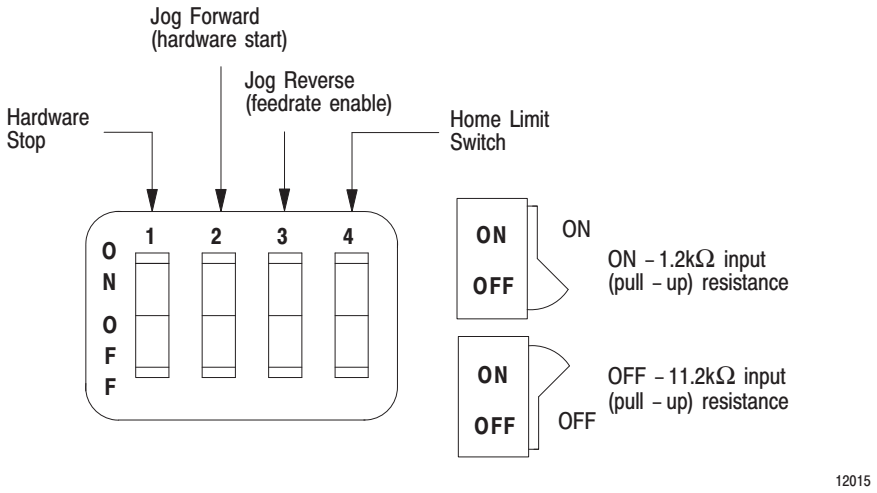


12014

Selecting Discrete Input Resistance

Select the resistance between each discrete input terminal and the high side of the input power supply. To select 1.2k ohms, set the switch on. To select 11.2k ohms, set the switch off. (Figure 6.3)

Figure 6.3
Discrete-input-resistance Switch Assembly



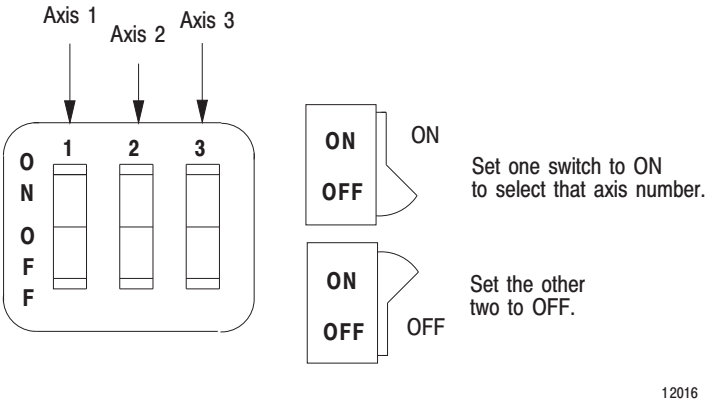
With 1.2k ohms, your input device must sink 4mA for a 5V power supply to 25mA for a 30V power supply. With 11.2k ohms, your input device must sink 0.5mA for a 5V power supply to 2.7mA for a 30V power supply.

Unless your input device cannot sink enough current, select 1.2k ohms because it provides better noise immunity than an 11.2k ohms input resistance.

Selecting Axis Number

Select the axis number as shown in Figure 6.4.

Figure 6.4
Axis-number Switch Assembly



Set to on the switch corresponding to the number for the axis. Set to off the other two switches in the assembly. Set each 1771-ES expander in an I/O chassis to a unique axis number, starting with 1.

Selecting Encoder Input Polarity

Select the polarity of each encoder input to allow your encoder to function properly with the 1771-ES expander (Figure 6.1).

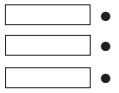
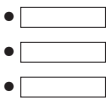
Encoder Polarity	Jumper Position
High–true	Left
Low–True	Right

With a differential encoder, the connections and the polarity jumper positions determine the polarity of the feedback signals. With a single-ended encoder, the polarity jumper positions alone determine the polarity of the feedback signals.

The polarity selections are important to the marker logic. Set the polarity so that the marker is true at the same time that channels A and B are true (refer to Figure 3.7)

Selecting Encoder Input Signal Mode

Select the signal mode of each encoder input to match the encoder (Figure 6.1).

Encoder Signal Mode	Jumper Position
Single–ended	Left 
Differential	Right 

Selecting Marker Logic

For almost all encoders, set the marker logic jumper to the bottom position to gate the marker with channel A and channel B. This gives the marker signal a level of noise immunity.

However, if you cannot select the polarity so that the marker on your encoder is always true at the same time as the channel A and B signals, set the market logic jumper to the top position.

Keying

A package of plastic Keys (cat. no. 1771-RK) is provided as standard with each I/O chassis. When properly installed, these keys can guard against the seating of all but a selected type of module in a particular I/O chassis module slot. Keys also help align the module with the backplane connector.

Each module is slotted at the rear edge. Position the keys on the chassis backplane connector to correspond to these slots to allow the seating of the module.

Insert keys into the upper backplane connectors. Position the keys between the numbers at the right of the connectors. Refer to Figure 6.5 for the 1771-M3 controller keying position. Refer to Figure 6.6 for the 1771-ES expander keying positions.

Figure 6.5
Keying Diagram for the 1771-M3 Controller

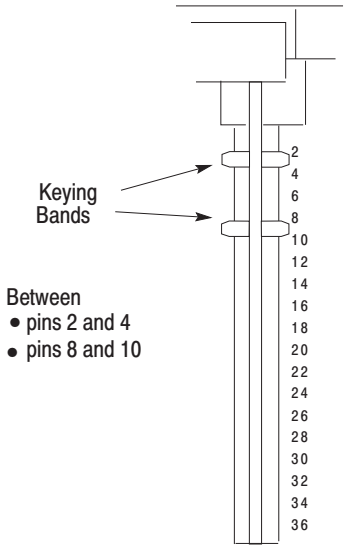
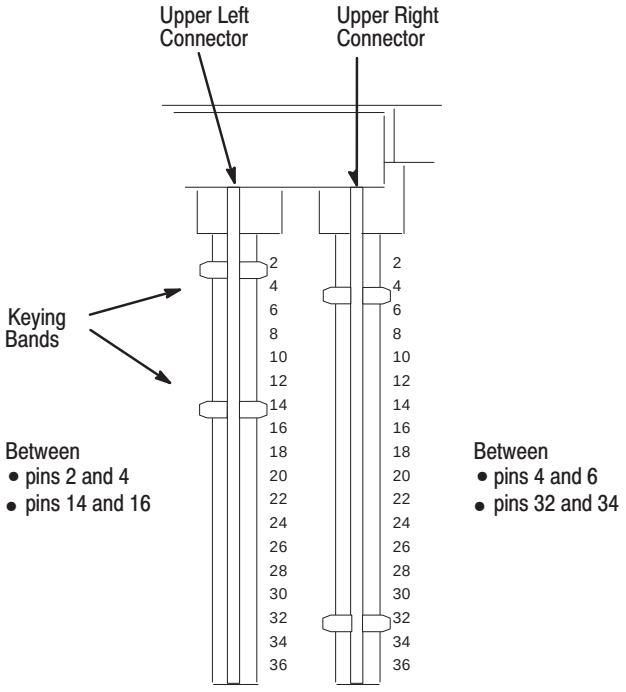


Figure 6.6
Keying Diagram for the 1771-ES Expander


Upper Left Connector
Upper Right Connector
Keying Bands
Between
• pins 2 and 4
• pins 14 and 16
Between
• pins 4 and 6
• pins 32 and 34
2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36
2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36
11006

Inserting the Module

To insert a module into an I/O chassis, follow these steps:

1. Remove power from the I/O chassis before inserting or removing a module.

2. Open the module locking latch on the I/O chassis and insert the module into the slot keyed for it.

3. Press the module firmly to seat it into its backplane connector.

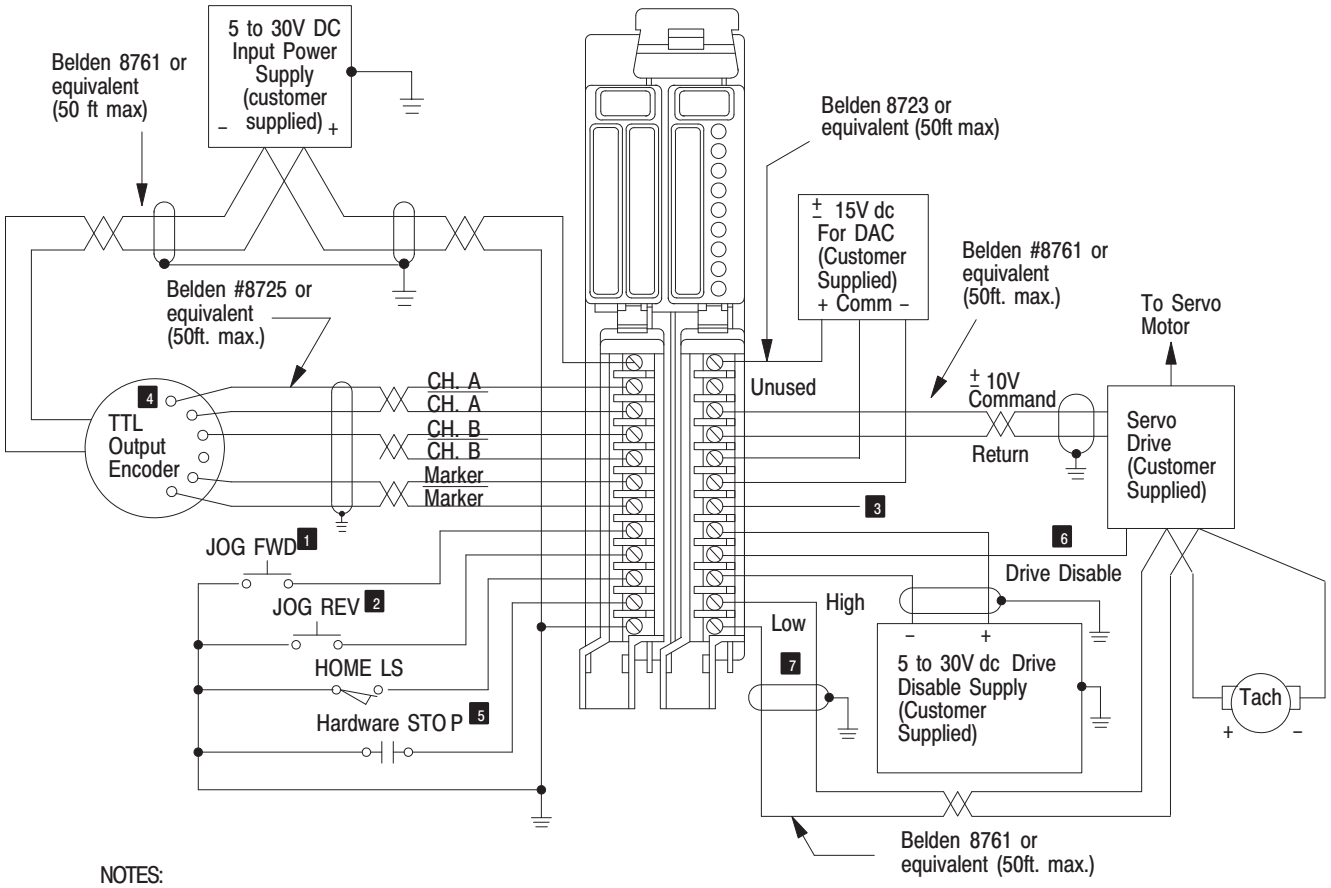
4. Secure the module in place with the module locking latch.

CAUTION: Do not force a module into a backplane connector; if you cannot seat a module with firm pressure, check the alignment and keying. Forcing a module can damage the backplane connector or the module.

Connecting to Terminals

Make connections to the 1771-ES expander as shown in Figure 6.7.

Figure 6.7
Simplified I/O Terminal Connection Diagram



NOTES:

If equipment permits, one supply can be used for encoder and input circuits. Current requirements depend on hardware configuration.

1. In the auto mode, the module accepts this input as the hardware start signal (figure 6.9).
2. In the auto mode, the module accepts this input as the feedrate enable signal (figure 6.9).
3. The module generates a hardware done signal at this +15V dc driver output terminal (figure 6.12).
4. Refer to figures 6.10 and 6.11.
5. Refer to figure 6.8.
6. Refer to figures 6.13 and 6.14.
7. Refer to figure 6.15.

12017

This is a simplified diagram to give you an overall view of how you are to connect these terminals. We give you further details in the following sections and their associated figures. For all connections to the terminals, limit the cable length to 50 feet.

Keep low-level conductors separate from high-level conductors. This is particularly important for cable connections to the encoder. Follow the practices outlined in the PC Grounding and Wiring Guidelines (publication 1770-980).

Power Supplies

Use shielded cable for connecting the input power supply and the analog power supply. Route these cables only with low-level conductors. Keep these power supply cables as short as possible. Ground the common terminal for each of these power supplies.

Encoder and Tachometer

For an encoder or tachometer connection, use only a single, continuous, shielded cable segment. Do not break the cable for connection in a junction box. Connect the cable directly from the encoder to the 1771-ES expander.

Important: Ensure that the power supply for the encoder provides the voltage recommended by the encoder manufacturers.

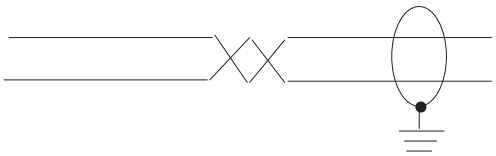
Shielded Cables

For many connections, we tell you to use shielded cable. Using shielded cables and properly connecting their shields to ground protects against electromagnetic noise interfering with the signals transmitted through the cables.

WARNING: Use shielded cable where we tell you to use it and how we tell you to use it. If you do not, the axis motion in your positioning system could be unpredictable; this could result in damage to equipment and/or injury to personnel.

Within a shielded cable, pairs of wires are twisted together. Using a twisted pair for a signal and its return path provides further protection against noise. We show a twisted pair like this:

We show a shielded twisted pair like this:



Connect each shield to ground at one end only. At the other end, cut the shield foil and drain wire short and cover them with tape to protect against their accidentally touching ground. Keep the length of leads extending beyond the shield as short as possible.

Use cables with the proper number of individually shielded twisted pairs as follows:

To connect to:	Number of Individually Shielded Twisted Pairs:	Use:
Encoder	4	Belden 8725 or equivalent
Analog power supply	2	Belden 8723 or equivalent
All other shielded cable connections	1	Belden 8761 or equivalent

Connecting the Input Supply

To connect the input power supply, follow these steps:

1. Connect the plus side of the input power supply to terminal 1 of the left wiring arm.

2. Connect the minus side to terminal 12 and to ground at the I/O chassis.

3. Connect the shields of the two cable segments if you use the same supply to power the encoder.

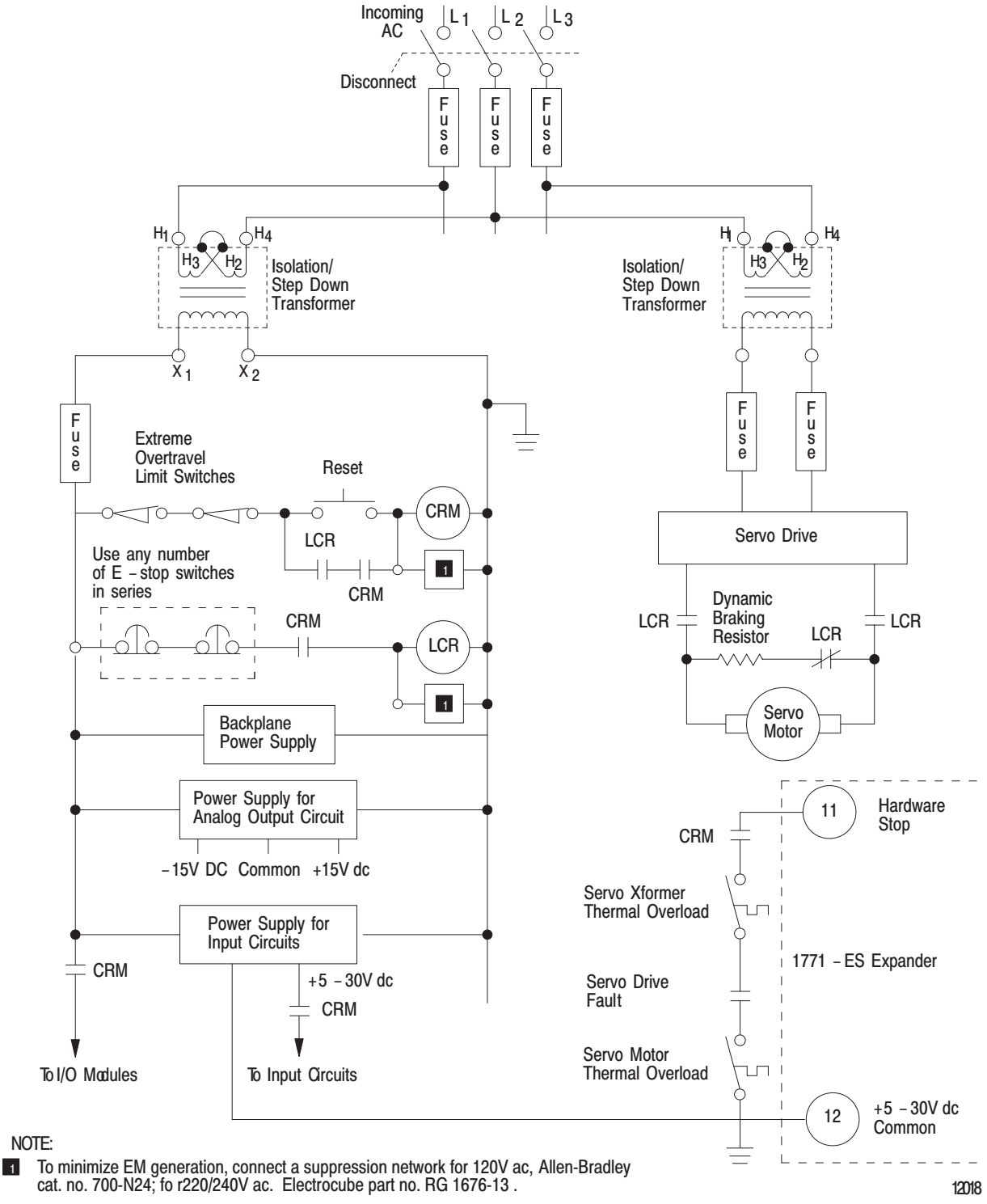
4. Connect the shield to ground at the I/O chassis end.

5. Connect the power-supply chassis to ground.

Connecting Hardware Stop

Before you connect to the hardware stop input, you should first consider overall power distribution, including the master-control relay and loop-contactor relay (Figure 6.8). Connect a suppression network across each relay coil.

Figure 6.8
Simplified Power Distribution with the Master-Control Relay, Loop-contactor Relay, and Hardware Stop



NOTE:

1 To minimize EM generation, connect a suppression network for 120V ac, Allen-Bradley cat. no. 700-N24; fo r220/240V ac. Electrocube part no. RG 1676-13 .

12018

Provide one transformer for the master-control relay (CRM) circuit, the loop-contactor relay (LCR) circuit, the dc power supplies, and any ac I/O chassis. Provide a separate transformer for the servo drives to provide noise immunity.

Use normally-open LCR contacts to switch power from the servo drive to the servo motor. Also, use normally closed LCR contacts to switch in the dynamic braking resistor across the servo motor whenever power is removed from the servo motor. Check with the servo drive and servo motor manufacturer for the resistance and power rating for the dynamic braking resistor.

WARNING: Without a dynamic braking resistor, removing servo motor power while the axis is in motion allows momentum to keep the axis in motion. In an emergency situation, this could be dangerous. A dynamic braking resistor can help stop the servo motor by quickly dissipating the energy of momentum. Even with dynamic braking, a vertical axis may also require an electric brake or counter balance.

An extreme overtravel limit switch or an emergency stop switch can de-energize the LCR, thereby turning off servo motor power. However, abruptly stopping an axis in this way stresses the servo motor and the mechanical linkage. Therefore, use the LCR to stop a moving axis only in an emergency. To stop an axis in a non-emergency situation, use the slide-stop bit in the command block thru the ladder diagram program. A slide stop decelerates the axis feedrate before stopping it. After a slide stop you can use an emergency stop switch if you want to remove power.

Connect a set of normally-open CRM contacts in series with servo transformer overload, servo drive fault, and servo motor overload contacts. Connect this series of contacts between the hardware stop input terminal and common. The opening of any of these contacts indicates that power to the servo motor is interrupted. When any of these sets of contacts open the hardware stop circuit, the following occur:

1. When this circuit opens, the 1771-ES expander immediately sets the velocity command output to zero and disables the serve drive by turning off the drive disable circuit.

2. The 1771-M3 controller sends the hardware stop signal to the PC data table thru the status block transfer.

3. After this circuit closes again, the 1771-ES expander still holds the velocity command at zero and holds the servo drive disabled until you either:

   - send a reset signal through a command block transfer (This allows the 1771-ES controller to maintain the accumulated axis position.)
   - cycle I/O chassis backplane power off, then back on (This clears the accumulated axis position.)

When you restart the axis after a hardware stop, the axis feedrate accelerates before reaching the final velocity rate. This allows a smooth start-up after a hardware stop.

Do not provide switch contacts in the hardware stop circuit for an operator to turn off the axis motion. Opening the hardware stop circuit stops the axis abruptly, stressing the servo drive, the servo motor, and the mechanical linkage, just as the CRM would. Use the hardware stop input only for backup to inform the 1771-ES expander of a condition that has already stopped the axis so that the expander can provide a controlled start-up.

Connecting Home Limit Switch

To connect a home limit switch, follow these steps:

1. Connect a normally open limit switch between the home limit switch terminal and common.

2. Place the limit switch so that it closes as the axis reaches a point approximately one half of an encoder revolution from the point you want to establish as home position.

3. Adjust the angular position of the encoder to have the marker pulse occur precisely at the point you want to establish as home position.

Through the command block transfer you can command a search home function (sections titled “Axis Control Word” and “Axis Control Word 2”). The 1771-ES expander:

- moves the axis to the limit switch
- decelerates the axis

- establishes the point of the next marker pulse following the limit switch as the home position
- stops the axis at the home position

You must re-establish the home position after each time power to the I/O chassis backplane goes off, because the encoder feedback is incremental, .

Connecting Jog Reverse (Feedrate Override Enable)

Figure 6.9 shows details of how to connect jog reverse and feedrate override enable. Follow these steps:

1. Provide a 3-pole selector switch to select between auto and manual mode.

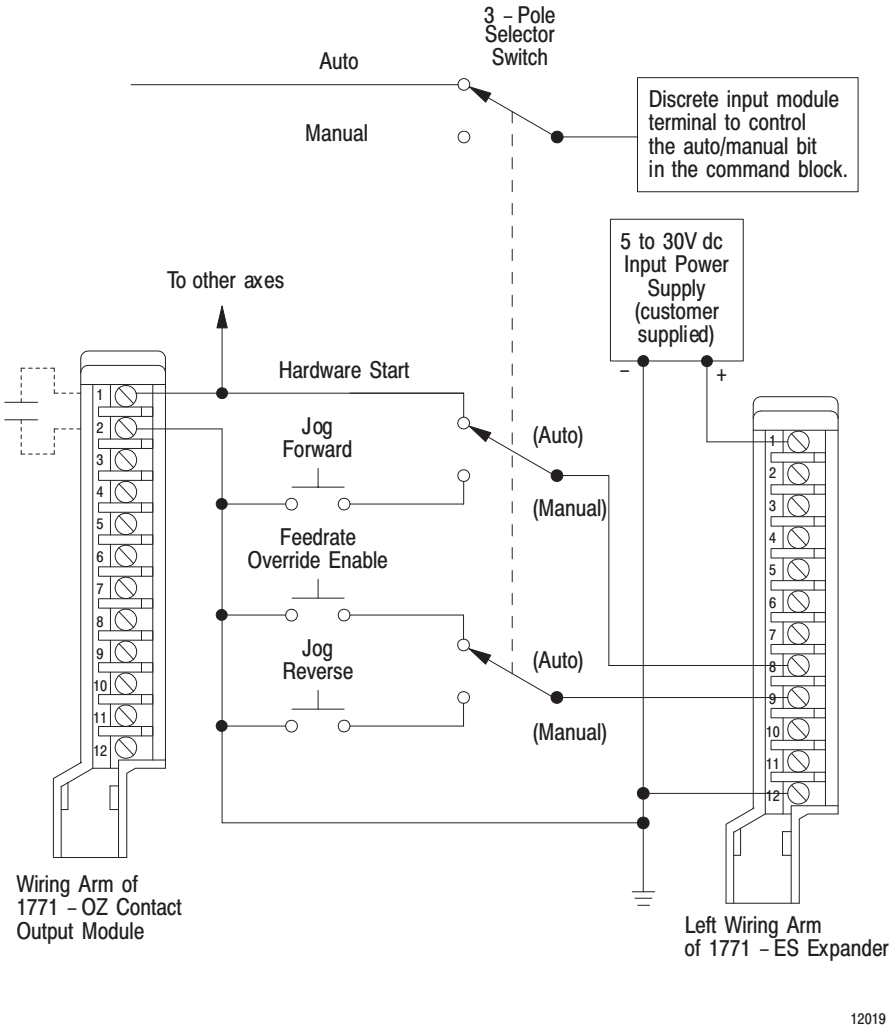
2. Connect one pole of the selector switch to a discrete input module terminal. Use this input to control the auto/manual bit in the control block. This bit controls whether the 1771-ES expander is in the auto or manual mode.

3. Connect a second pole of the selector switch to the jog reverse (feedrate override enable) terminal of the 1771-ES expander.

4. Connect a momentary-contact jog reverse switch to the selector switch contact corresponding to manual on the second pole.

5. Connect a momentary-contact feedrate override switch to the selector switch contact corresponding to auto on the second pole.

Figure 6.9
Connection Details for Jog Forward (Hardware Start) and Jog Reverse (Feedrate Override Enable)



In the manual mode, the jog reverse switch controls whether the input is high or low. In the auto mode, the feedrate override enable switch controls whether the input is high or low. You can connect the same feedrate override enable signal to several 1771-ES expanders to coordinate the start of feedrate override for those axes.

Connecting Jog Forward (Hardware Start)

Figure 6.9 also shows details of how to connect jog forward and hardware start. Follow these steps:

1. Connect a third pole of the selector switch to the jog forward (hardware start) terminal of the 1771-ES expander.

2. Connect a momentary-contact jog forward switch to the selector switch contact corresponding to manual on the third pole.

3. Connect an output terminal of a Contact Output Module (cat. no. 1771-OZ) to the selector switch contact corresponding to auto on the third pole.

In the manual mode, the jog forward switch controls whether the input is high or low. In the auto mode, the hardware start output from the 1771-OZ module controls whether the input is high or low.

You can use the ladder diagram program to generate a hardware start signal (by closing the contacts of 1771-OZ module output) when each of several axes generates a hardware done signal. You can connect the same hardware start signal to several 1771-ES expanders to coordinate the start of motion following halt moves for these axes.

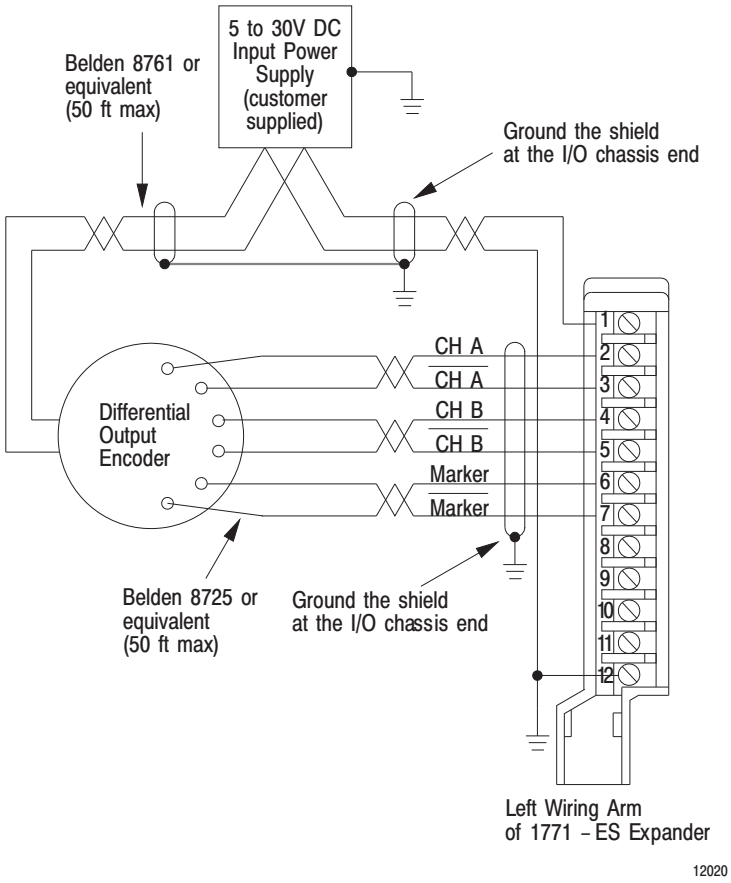
Connecting a Differential Encoder

Figure 6.10 shows details of how to connect a differential encoder. With a differential encoder, reversing the connections on a channel or changing the position of the polarity jumper for the channel reverses the polarity of the signal on that channel. Set the polarity so that the marker is true at the same time that channels A and B are true.

If you switch channel A with channel B, you reverse the direction of the feedback. If the direction of the feedback does not correspond to the axis motion direction, as you have defined it, switch channel A with channel B.

Ground the shield at the I/O chassis end.

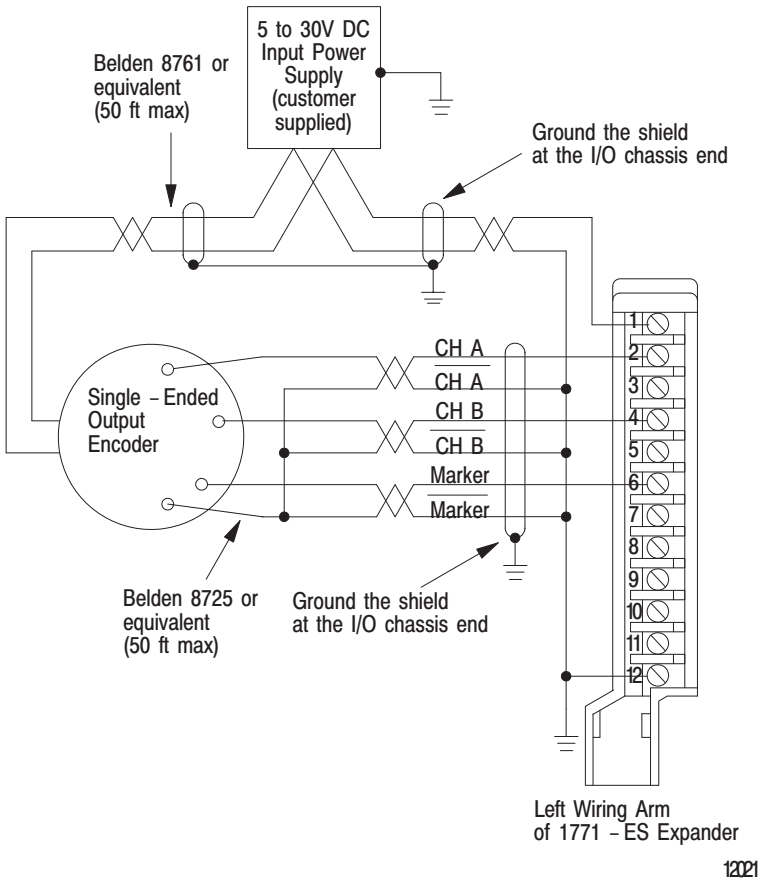
Figure 6.10
Connection Details for a Differential Encoder



Connecting a Single-Ended Encoder

Figure 6.11 shows details of how to connect a single-ended encoder. Connect each channel return line to common.

Figure 6.11
Connection Details for a Single-ended Encoder



If you switch channel A with channel B, you reverse the direction of the feedback. If the direction of the feedback does not correspond to the axis motion direction, as you have defined it, switch channel A with channel B.

Ground the shield at the I/O chassis end.

Connecting the Analog Output Supply

To connect the analog output supply, follow these steps:

1. Connect the plus (+) side of the analog and hardware one output power supply to terminal 1 of the right wiring arm.

2. Connect the minus (-) side to terminal 6.

3. Connect the common to terminal 5.

4. Connect the shield to ground at the I/O chassis.

Connecting Velocity Command

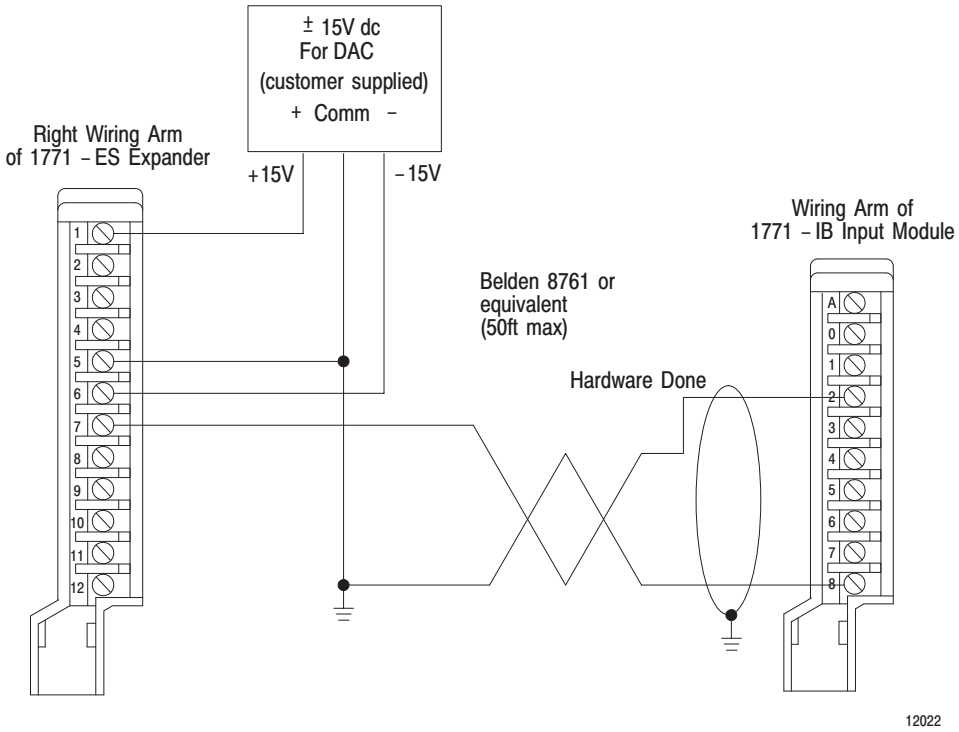
Connect the analog velocity command output signal from terminals 3 and 4 on the right wiring arm to the corresponding terminals of the servo drive. Reversing these connections reverses the direction the axis moves in response to the velocity command. Connect this signal so that the direction of motion that results from it matches the correct direction of motion as you have defined it.

Connect the shield to ground at the servo drive end.

Connecting Hardware Done

Figure 6.12 shows details of how to connect hardware done. Follow these steps:

Figure 6.12
Connection Details for Hardware Done Output



1. Connect the hardware done output from terminal 7 on the right wiring arm to an input terminal of a dc (12-24V) Input Module (cat. no. 1771-IB).

2. Connect the analog and hardware done output power supply common to the 1771-IB input module common terminal. This power supply provides the +15V dc source for the hardware done signal.

Examine the hardware done signal thru the ladder diagram program. You can synchronize the motion of several axes after each halt move: send a hardware start signal to all axes when you have received the hardware done signal from each axis.

Connecting Drive Disable

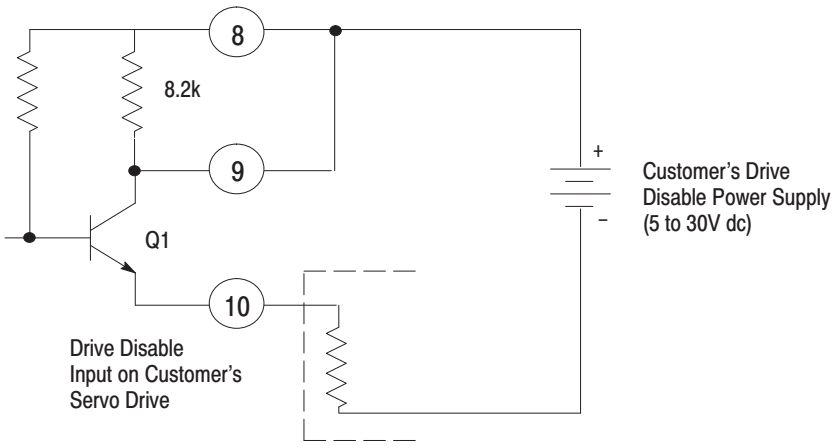
Figure 6.13 shows details of how to connect drive disable for two basic types of configurations. Some servo drives require a current source connected to an input to enable the drive. Some require a current sink connected to an input to enable the drive. We provide all three connection points (base, emitter, and collector) of the drive disable circuit to provide you with a flexibility of connecting it in a configuration that applies to your servo drive.

Figure 6.13
Connection Details for Two Basic Drive Configurations

a) Current Sourcing Configuration

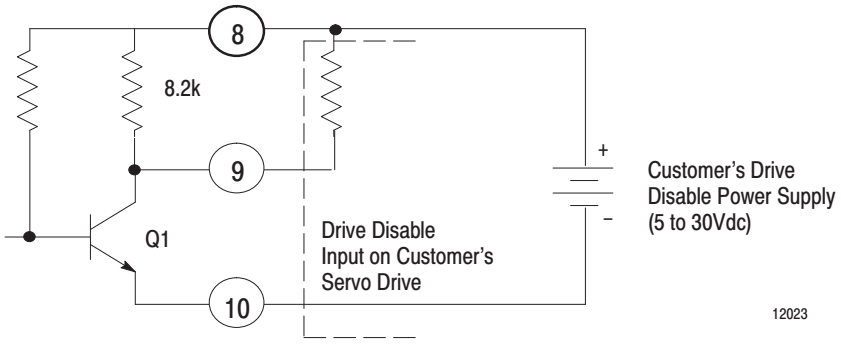
Drive Enable Q1 on: Current is sourced from terminal 10 into the servo drive.

Drive Disabled Q1 off: Current into the servo drive is inhibited.



b) Current Sinking Configuration
Drive Enable Q1 on: Current is sunk thru terminal 9 and Q1.

Drive Disabled Q1 off: Current thru Q1 is inhibited. Terminal 9 is pulled up to the potential of terminal 8.



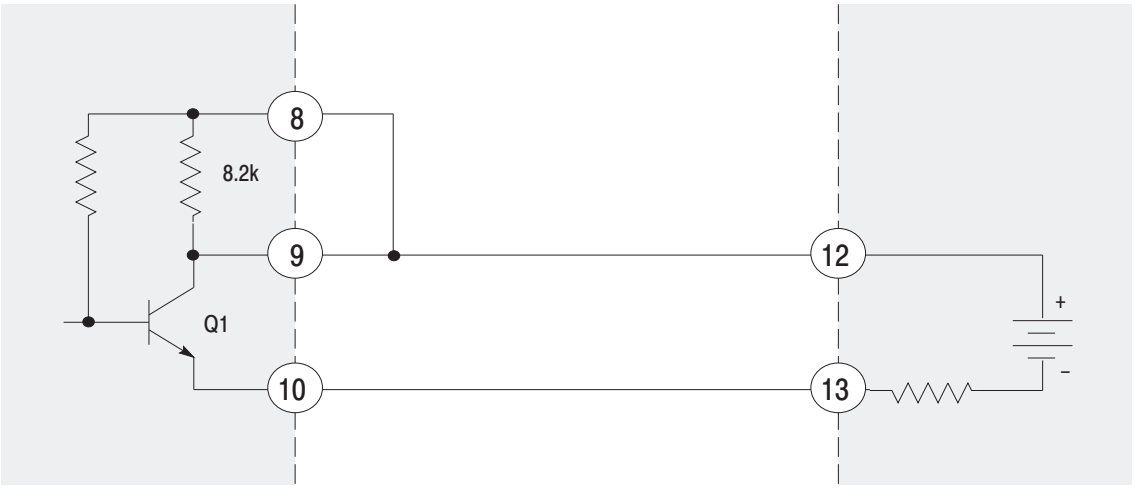
For the drive disable circuit, you must provide a 5-30V dc power supply which can provide 100mA maximum. The power supply can be separate or an integral part of the servo drive. Each of the configurations of figure 6.13 includes a separate power supply.

Figure 6.13a shows a current sourcing configuration. Normally the drive disable circuit is on, sourcing current into the drive thru terminal 10. When the drive disable circuit turns off, the drive is disabled.

Figure 6.13b shows a current sinking configuration. Normally the drive disable circuit is on, sinking current from the drive thru terminal 9. When the drive disable circuit turns off, the drive is disabled.

Figure 6.14 shows how to connect the drive disable circuit to the Bulletin 1388 servo drive which has an internal power supply and requires a current source to enable it.

Figure 6.14
Connection Details for Providing a Drive–disable Signal to the Bulletin 1388 Servo Drive

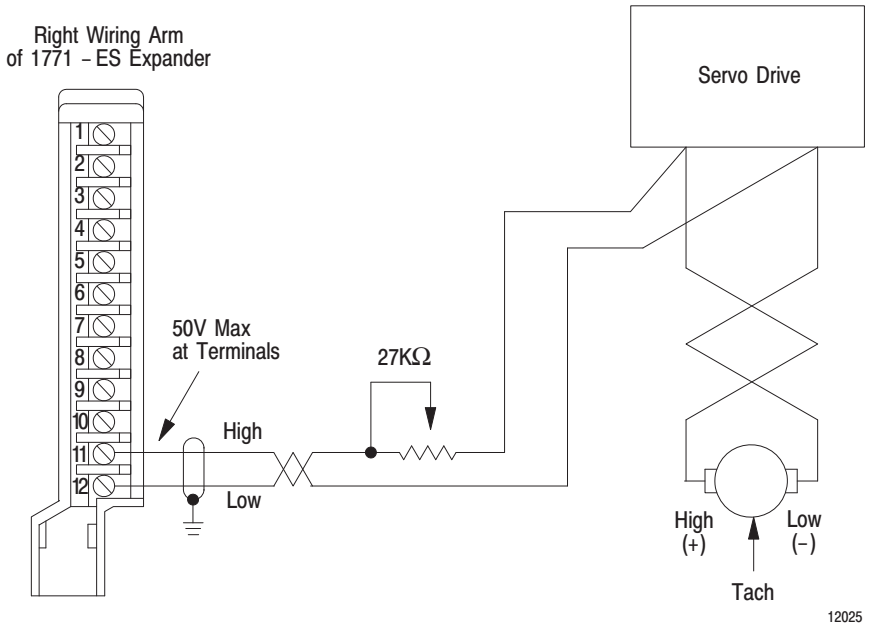


Note that whatever configuration your drive requires, you must connect the plus side of the power supply to terminal 8 on the right wiring arm of the 1771-ES expander. Without this connection, the drive disable circuit will not turn on; the 1771-ES expander will not enable the servo drive.

Connecting the Tachometer

Figure 6.15 shows details of how to connect the tachometer. Follow these steps:

Figure 6.15
Connection Details for Tachometer



1. Connect the tachometer directly to the servo drive.

2. Connect the tachometer signal at the servo drive to the right wiring arm of the 1771-ES expander. This allows the 1771-ES expander to detect loss of tachometer feedback at the servo drive. Limit the voltage at the terminals to 50V maximum. Tachometers typically generate much larger voltages than 50V at high speed. Therefore, you must drop the voltage thru a voltage divider.

3. Unless you have access to a voltage divider in the servo drive, place a 27k ohms 1/4 Watt potentiometer between the servo drive and terminal 11 of the 1771-ES expander.

4. Set the potentiometer for maximum resistance until you perform the integration procedures (chapter 9).

5. Connect the tachometer high signal to terminal 11.

6. Connect the tachometer low signal to terminal 12.

7. Connect the shields of the cable segments.

8. Connect the shield to ground at the I/O chassis end.

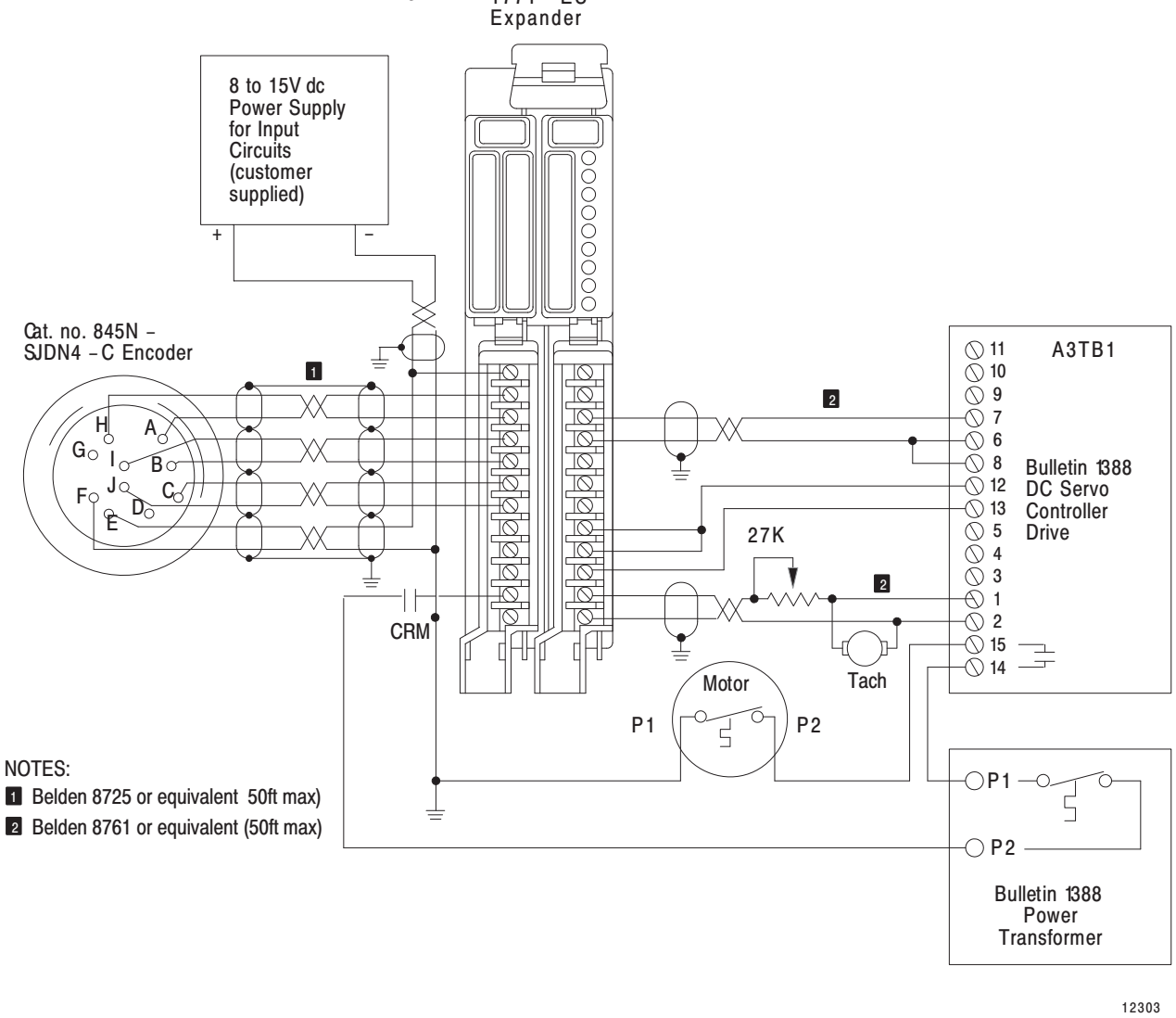
Connecting A-B Encoder and Drive

Figure 6.1 shows the jumpers in the position in which we place them for shipping the 1771-ES expander to you. These channel polarity jumper settings select high-true polarity. These channel signal mode jumper settings select differential mode. This marker logic jumper setting selects the marker to be gated with channel A and channel B. If you use the Allen-Bradley 845N-SJDN4-C encoder, leave the jumpers set to the position shown in Figure 6.1.

With the jumpers set as shown in figure 1, connect the 845N-SJDN 4-C encoder to the 1771-ES expander as shown in Figure 6.16. We show the **channel A** signal connection reversed with the **not channel A** connection and the **channel B** signal connection reversed with the **not channel B** connection. This inversion of the channel A and B polarity allows the marker to be high at a time when both channels A and B are high.

Use an 8 to 15V dc power supply for the input circuits. Connect the plus side of the supply voltage to pin E of the encoder. With this configuration, 5V dc power is generated at the encoder; the signals from the encoder are 5V dc.

Figure 6.16
Connections to a Cat. No. 845N-SJDN4-C Encoder and a Bulletin 1388 dc Servo Controller Drive



Connect the bulletin 1388 dc servo controller drive and its bulletin 1388 power transformer to the 1771-ES expander as shown in Figure 6.16. Tachometer input terminal 12 on the 1771-ES expander and terminal 2 on the drive each connect to a dc common at ground potential; therefore, you must connect these terminals directly as shown.

Connect the analog output signal from terminal 3 of the 1771-ES expander to terminal 7 of the drive. Connect the analog return signal from terminal 4 of the 1771-ES expander to terminals 6 and 8 of the drive. With this signal orientation, when you connect the tachometer to the drive with the proper polarity for negative feedback, the signal will also have

the proper polarity for loss-of-feedback detection at the 1771-ES expander. If you use the opposite analog output signal orientation, you will not be able to utilize the loss-of-feedback detection feature.

WARNING: Always utilize the loss-of-feedback feature. Without loss-of-feedback detection, if encoder or tachometer feedback is lost, unexpected axis motion can occur, resulting in damage to equipment and/or injury to personnel.

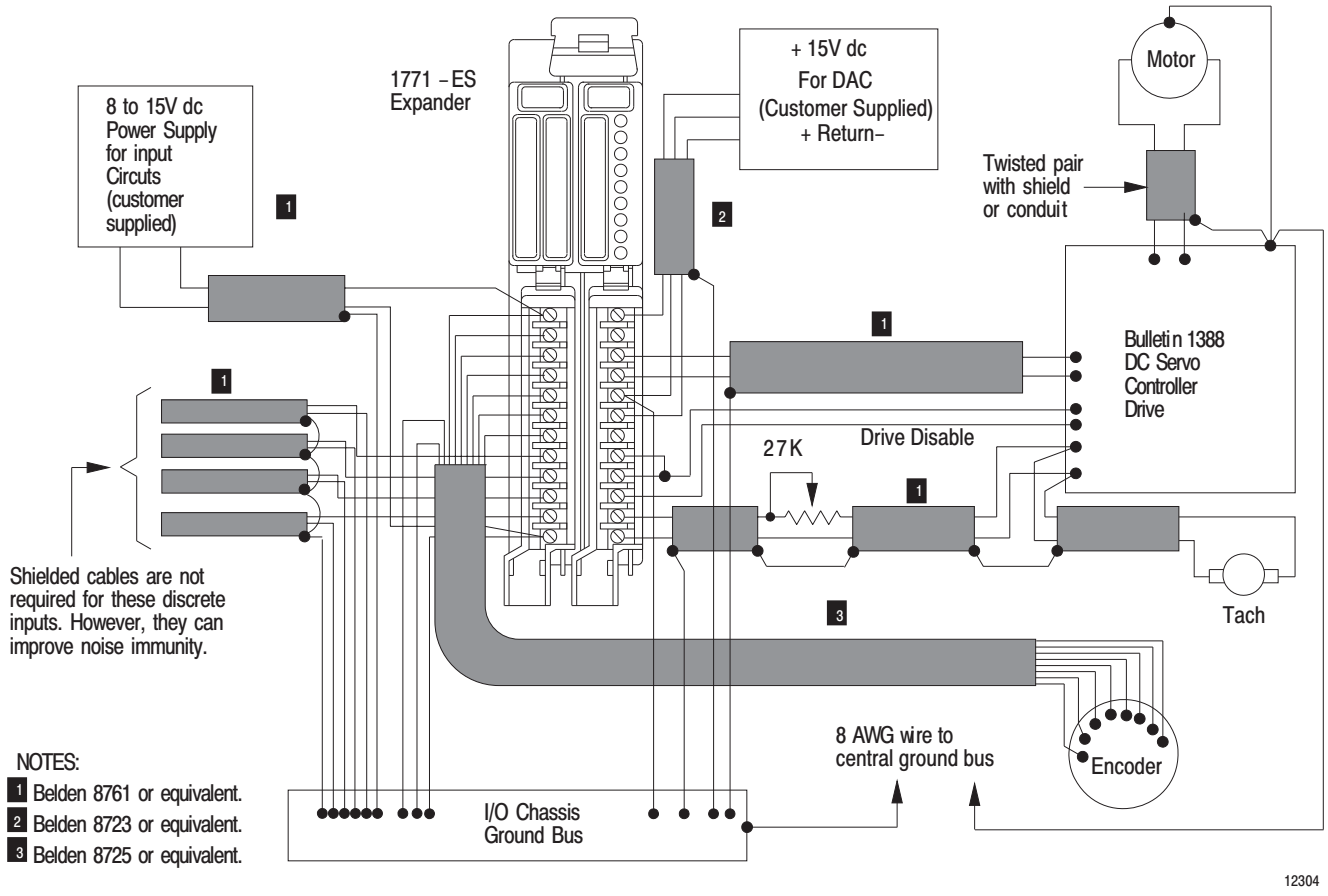
Limit the cable lengths to 50 feet. If your application requires a cable length greater than 50 feet, contact your local Allen-Bradley representative.

Grounding Cable Shields

Figure 6.17 is a pictorial representation of the shielded cable connections. Mount a ground bus directly below the I/O chassis to provide a connection point for cable shield drain wires and the common connections for the input circuits. Connect the I/O chassis ground bus through 8 AWG wire to the central ground bus to provide a continuous path to ground.

The tachometer cable is broken into three segments because of the connection to the drive and potentiometer in the middle of the cable. Connect these cable shield segments together as shown. Connect the shield to ground only at the I/O chassis end. Do not connect the shield to the drive.

Figure 6.17
Shielded Cable Grounding Connections



Connecting AC Power

Figure 6.18 shows ac power connections. Incoming ac connects to the primary of the bulletin 1388 power transformer. Both the 120V secondary and the 35.5V secondary connect to the bulletin 1388 dc servo controller drive.

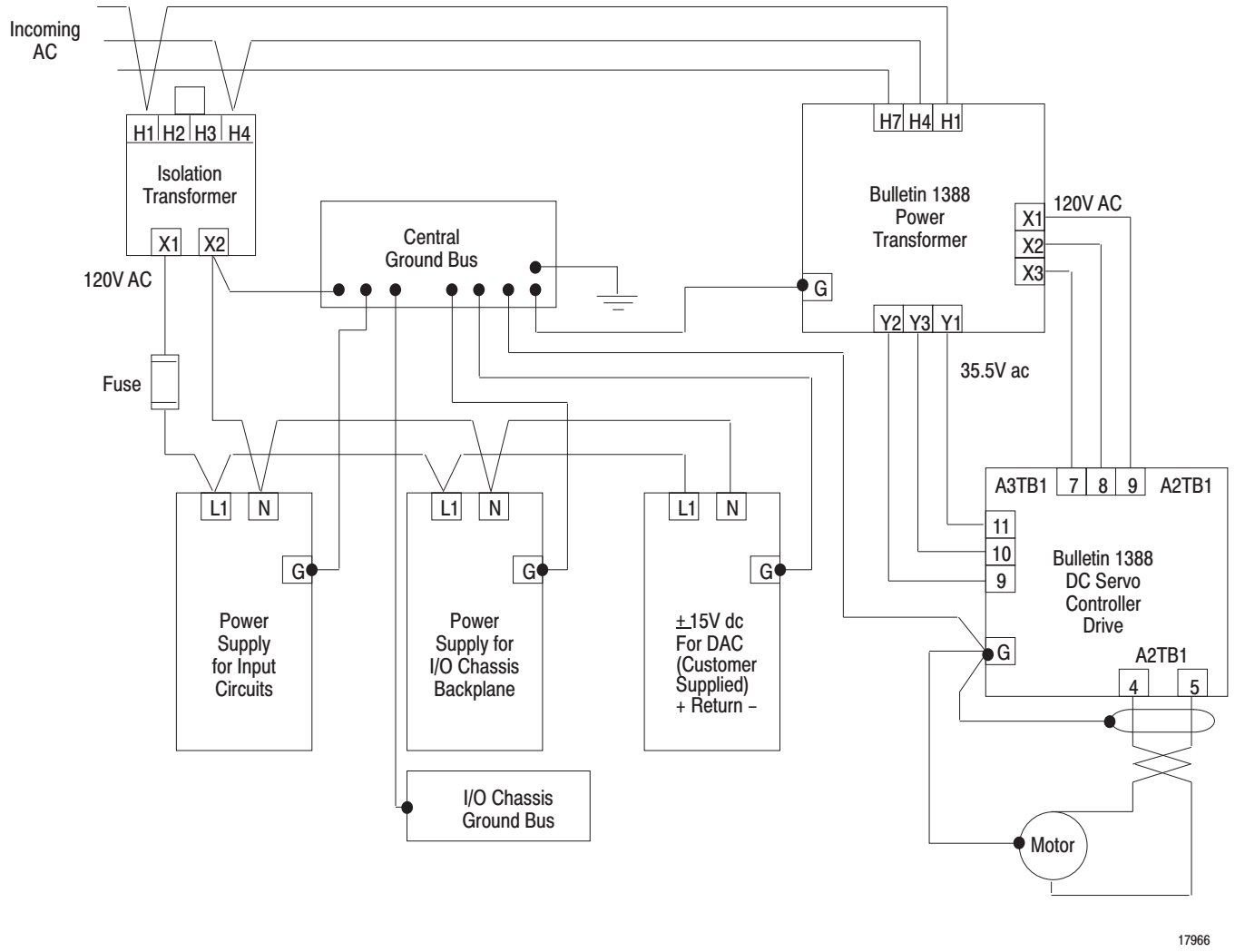
Incoming ac also connects to the primary of an isolation transformer. The secondary of the isolation transformer connects to:

- the power supply for the input circuits
- the power supply for the I/O chassis backplane
- the power supply for the analog output circuit

Figure 6.18 shows a grounded ac system; the low side of the isolation transformer is connected to the central ground bus. Figure 6.18 also

shows connections from the central ground bus to each chassis and to the I/O chassis ground bus shown in Figure 6.17.

Figure 6.18
AC Power and Ground Connections



Start-up Sequence

After properly installing your servo positioning assembly, formatting the data blocks, entering the program, and integrating each axis, you start up the system in the following sequence.

1. De-energize the CRM relay.

2. Turn on the dc power connected to the wiring arms.

3. Turn on the power supply for the I/O chassis backplane.

4. Energize the CRM relay.

5. Generate a reset command through the command block.

Summary

Now that you have installed the servo positioning assembly, you are ready to enter data blocks into the data table of the PC processor. During installation you made hardware selections to direct module operation. In chapter 7, we tell you how to make software selections to direct other aspects of module operation.

Chapter 7

Formatting and Interpreting Data Blocks

Chapter Objectives

The previous chapter told you how to install the modules. During installation, you made hardware selections through switch and jumper settings. These hardware selections direct some aspects of module operation.

This chapter tells you how to make software selections through data blocks you set up in the data table. Through data blocks you direct module operation.

This chapter also tells you how to monitor module operation through a data block that the module sends to the data table.

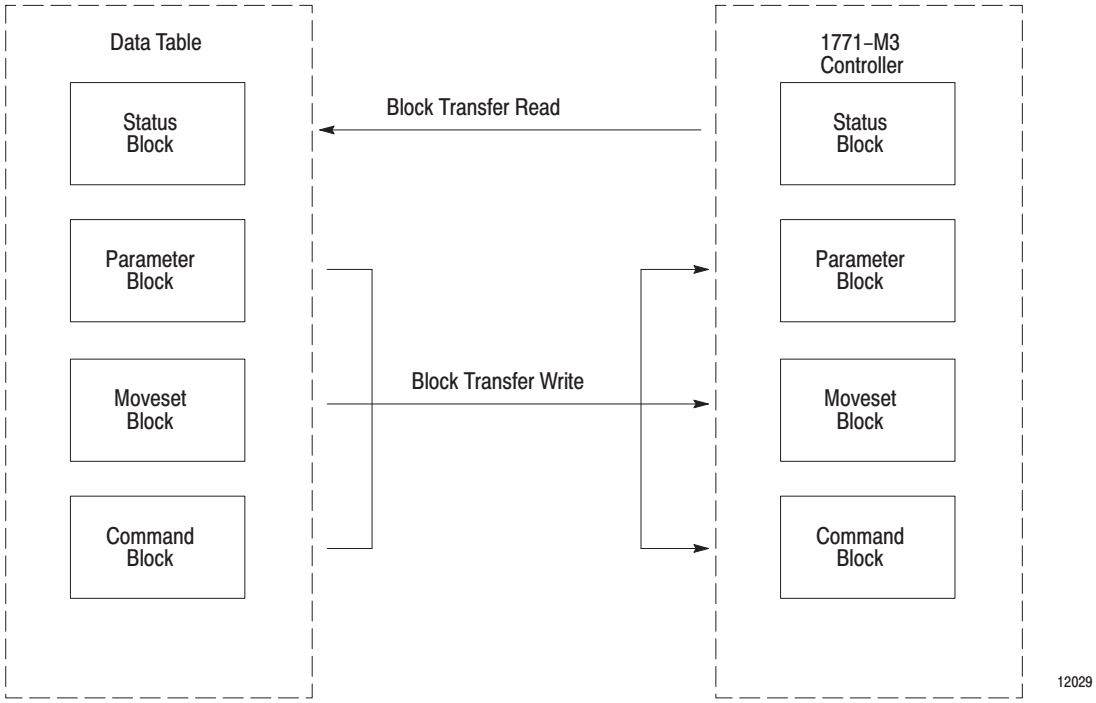
Relationship of Data Blocks

You must program the PC processor to communicate with the 1771-M3 controller through a block-transfer-read instruction and a block-transfer-write instruction. The data blocks are:

- status block
- parameter block
- moveset block
- command block

The block-transfer-read instruction transfers status block data from the 1771-M3 controller to the data table. The block-transfer-write instruction transfers the parameter block, the moveset block, and the command block data from the data table to the module. (Figure 7.1).

Figure 7.1
The Status Block Transfers to the Data Table - the Parameter, Moveset, and Command Blocks go to the 1771-M3 Controller



Status Block

The status block is regularly transferred to the data table to provide updated information about the current status of each axis. This status includes:

- actual axis position
- in position
- at home position
- slide stop
- emergency stop
- software travel limit exceeded
- feed reduction
- excess following error
- auto/manual mode
- address pointer to tell the program which block (parameter, moveset, or control) to write transfer to the 1771-M3 controller next
- diagnostic status that tells you where programming errors are in parameter, moveset, and command blocks

The first block transfer after power-up writes a 6-word status block into the data table. After that, the status block consists of 6 words for a 1-axis system, 10 words for a 2-axis system, or 14 words for a 3-axis system. You establish the address for the status block through the block transfer read instruction. Because axis command and status data is stored in the data table, axis motion control can interact with other axes, discrete I/O, and report generation.

Parameter Block

The parameter block for a 1-axis system has 25 words; a 2-axis system has 44 words; a 3-axis system has 63 words.

You specify parameters for each axis separately. You specify parameters such as:

- software travel limits
- home position
- servo gain
- global accel/decel rate
- rapid traverse rate

In the parameter block, you also specify the address of the parameter block, the command block, and the first moveset block for each axis. With these addresses, the 1771-M3 controller can ask (through the status block) for the block it needs at any particular time.

The processor transfers the parameter block to the 1771-M3 controller through a block transfer write. This provides axis parameter information after a power-up and after a command block commands a reset or new parameters.

Moveset Block

A moveset block describes a sequence of axis moves. You can program axis motion to provide either single-step moves or continuous moves.

Each move requires a minimum of three words (a single-move control word and two words to define position or dwell time) and can include three optional words (a rate word, an accel word, and a decel word) for a total of six. A moveset control word applies to the entire block. If additional moveset blocks are needed, you also need a next-moveset-point word. A moveset block can be 64 words long maximum and describe 21 moves maximum. To describe 21 moves in a single moveset block, all 21

moves would have to use the global accel/decel and final rate values from the parameter block.

Upon request from the status block, the PC processor sends a moveset block to the 1771-M3 controller, which transfers each move description to the 1771-ES expander one at a time. The 1771-ES expander generates the analog voltage to command axis motion as programmed.

Command Block

The command block for a 1-axis system has up to four words; a 2-axis system has up to eight words; a 3-axis system has up to 12 words. This block regularly transfers from the data table to provide commands (such as start, slide stop, search home, jog, reset and offset) for each axis unless the 1771-M3 controller needs a parameter or moveset block. You must include the command block address in the parameter block.

Data Table Allocation

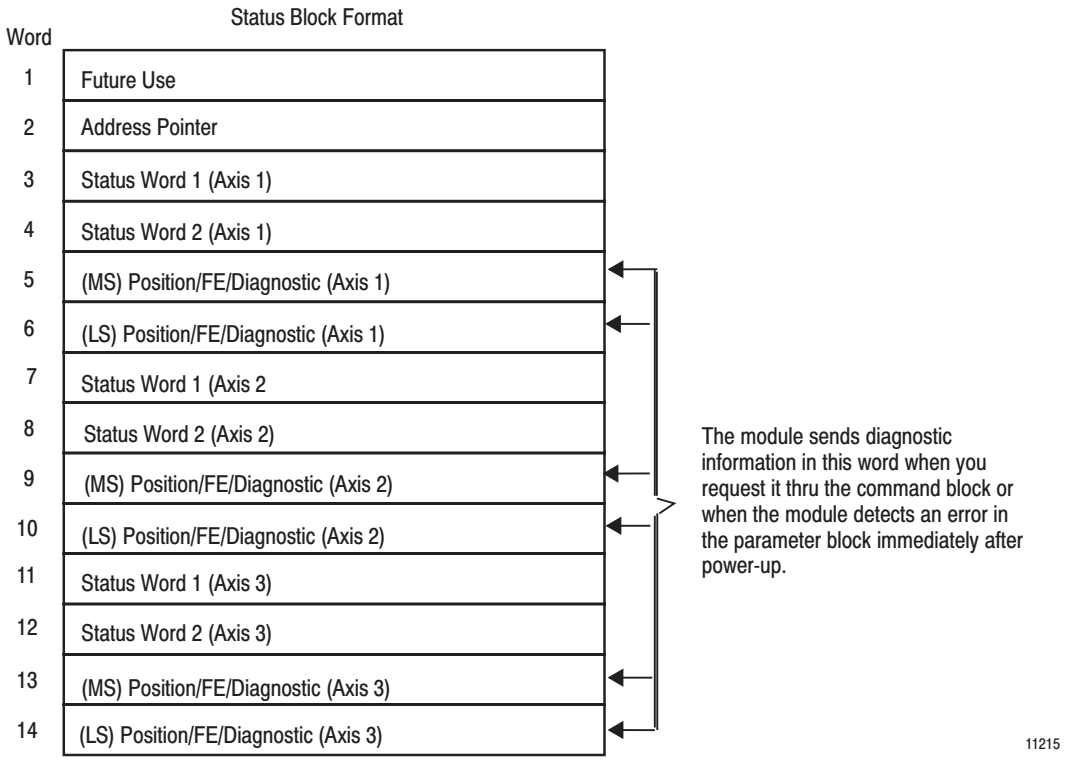
You must allocate a sufficiently large data table area for the data blocks needed in the block transfer communication. Furthermore, the parameter block must start at least 63 words before the end of a contiguous data table area. Also, each moveset block (regardless of size) must start at least 64 words before the end of a contiguous data table area. For a PLC-2 family processor, assign data block addresses of 200 or greater to avoid processor work areas.

Status Block

The status block, which is the only block transferred from the 1771-M3 controller to the processor, contains information about axis and servo positioning assembly status. The first block transfer after power-up writes a 6-word status block into the data table. After that, the status block consists of word assignments (Figure 7.2):

Number of Axes	Size of Status Block
1	6 words
2	10 words
3	14 words

Figure 7.2
Status Block - Showing Word Assignments


Status Block Format
Word
1 Future Use
2 Address Pointer
3 Status Word 1 (Axis 1)
4 Status Word 2 (Axis 1)
5 (MS) Position/FE/Diagnostic (Axis 1)
6 (LS) Position/FE/Diagnostic (Axis 1)
7 Status Word 1 (Axis 2
8 Status Word 2 (Axis 2)
9 (MS) Position/FE/Diagnostic (Axis 2)
10 (LS) Position/FE/Diagnostic (Axis 2)
11 Status Word 1 (Axis 3)
12 Status Word 2 (Axis 3)
13 (MS) Position/FE/Diagnostic (Axis 3)
14 (LS) Position/FE/Diagnostic (Axis 3)
The module sends diagnostic information in this word when you request it thru the command block or when the module detects an error in the parameter block immediately after power-up.
11215

We reserve the first word of the status block for future use. It contains all zeros when returned by the 1771-M3 controller. The second word is an address pointer that identifies the next block the processor is to transfer to the 1771-M3 controller. Words 3 thru 6 provide the status of axis 1. Words 7 thru 10 provide the status of axis 2. Words 11 thru 14 provide the status of axis 3.

The following sections describe status block words.

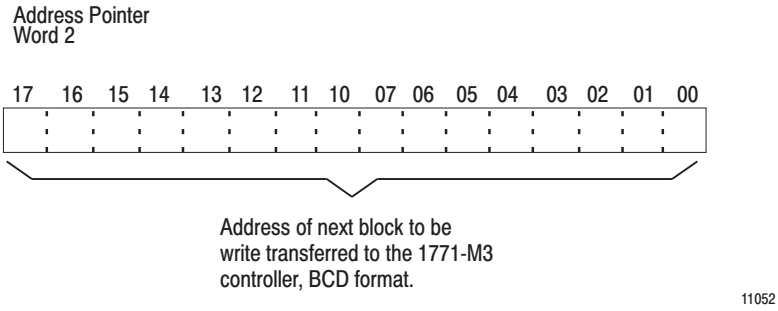
The servo positioning assembly configures all words in the status block.

Address Pointer

The address pointer word (Figure 7.3) contains, in BCD format, the data table address of the next block to be transferred from the processor to the 1771-M3 controller. Your ladder diagram program reads this address and uses it to configure a write block transfer instruction. The 1771-M3 controller programs this word according to its requirements. When it does

not need to request the parameter block or a moveset block, it requests the command block.

Figure 7.3
Address Pointer Word


Address Pointer
Word 2
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Address of next block to be
write transferred to the 1771-M3
controller, BCD format.
11052

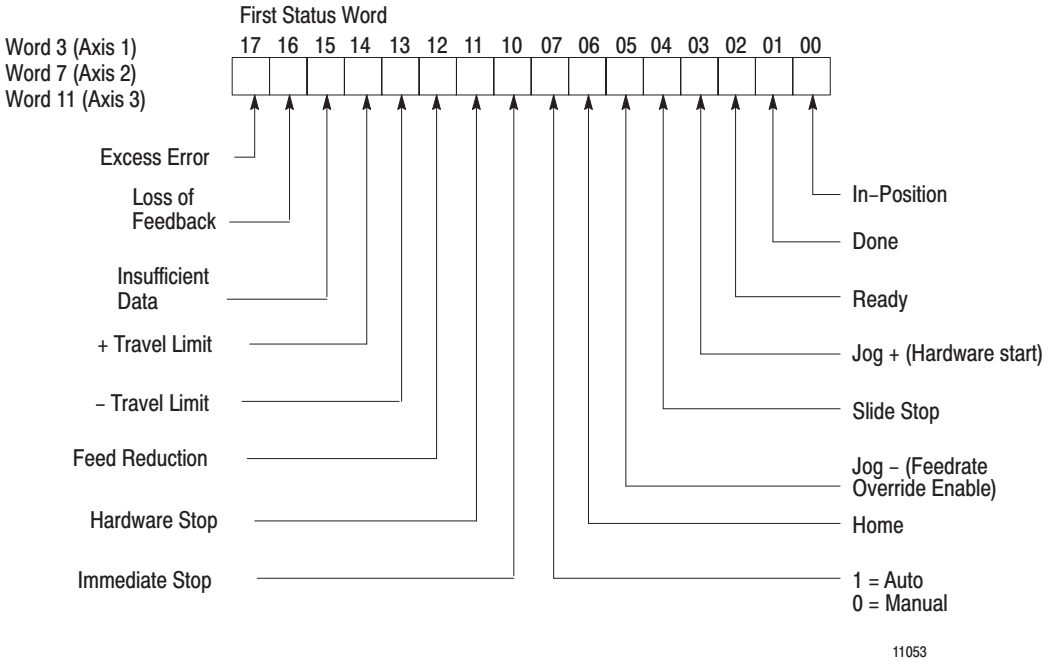
The value that appears in this word is one of the pointer addresses you put into:

- word 2 (parameter block) of the parameter block
- word 3 (command block) of the parameter block
- word 4 (initial moveset block, axis 1) of the parameter block
- word 5 (initial moveset block, axis 2) of the parameter block
- word 6 (initial moveset block, axis 3) of the parameter block
- the last word (next moveset block) of a moveset block

First Status Word

Each bit of the first status word (Figure 7.4) corresponds to a particular axis condition.

Figure 7.4
First Status Word



Bit 0 In-Position

The 1771-M3 controller turns on this bit when following error is less than twice the in-position band value programmed in the parameter block (word 11). When the in-position bit is on, it indicates that the axis has moved to within a specified distance of the programmed end point.

Bit 1 Done

The 1771-M3 controller turns on this bit when the 1771-ES expander has finished feeding the axis for a programmed move or finished a dwell.

Bit 2 Ready

The 1771-M3 controller turns off the ready bit after power-up or after you execute the reset command. The controller turns on this bit when it receives valid parameter-block values. When the ready bit is on, the 1771-M3 controller is ready to respond to commands you issue through the command block.

The processor must not transfer the command or moveset blocks to the servo controller until the ready bit is on.

Bit 3 Hardware Jog + (Hardware Start)

The 1771-M3 controller turns on this bit when the 1771-ES expander recognizes a jog plus or hardware start input signal.

Bit 4 Slide Stop

The 1771-M3 controller turns on this bit when it receives a slide-stop request from the command block (word 1, bit 5). The slide stop status bit stays on even after the slide-stop command is no longer present in the command block. This bit turns off when you command axis motion or reset. A reset command while the axis is in motion will also turn on this bit and cause a slide stop. When the axis stops, this bit turns off.

Bit 5 Hardware Jog - (Feedrate Override Enable)

The 1771-M3 controller turns on this bit when the 1771-ES expander recognizes a jog minus or feedrate override enable input signal.

Bit 6 Home

The 1771-M3 controller turns on this bit when the axis feed is done after any command to move to the home position, if you have established a home position. You establish a home position through an initialize home, or search home command. This bit turns off when the axis moves away from the home position.

Bit 7 Auto/Manual

This bit indicates the current mode of the axis, based on the status of the auto/manual bit (word 1, bit 7) in the command block (1=auto, 0=manual).

Bit 10 Immediate Stop

When this bit is on it indicates that the 1771-ES expander is holding its analog output signal at zero and is disabling the servo drive through its drive disable output. You can clear this immediate stop condition through a reset command or by cycling I/O chassis backplane power off, then on. Commands and events that can cause the immediate stop condition are:

- software stop command
- hardware stop input open
- excess following error
- loss of feedback
- loss of power
- firmware or hardware watchdog timeout on the 1771-ES expander

Bit 11 Hardware Stop

The 1771-M3 controller turns on this bit only if the hardware stop input of the 1771-ES expander is open. Note that the immediate stop bit (bit 10) is also on if this bit is on. You can turn off this bit with a reset command or by cycling power to the I/O chassis backplane.

Bit 12 Feed Reduction

This bit goes on when axis following error reaches 106.25% of rapid traverse following error, resulting in 50% feedrate reduction, but has not necessarily reached the excess error point. When axis following error does reach the excess error point, the feed reduction bit stays on, and the immediate stop status bit goes on.

Important: If the excess error point is less than 106.25% of rapid traverse following error, immediate stop occurs before feed reduction. Consequently, the feedrate reduction bit in the status block does not turn on.

Bit 13, 14 + and - Travel Limits

These bits are on when the axis is at the corresponding software travel limit positions. You enter the travel limits in the parameter block.

Bit 15 Insufficient Data

When the servo positioning assembly receives a command to execute axis motion, such as start or begin, but does not have moveset data to execute a move, it turns on the insufficient data bit. It also turns on this bit when you issue an escape command, even though you had never stored an escape move on the 1771-ES expander.

This insufficient data bit stays on until the 1771-M3 controller receives a new moveset block and then a start or begin command.

Bit 16 Loss of Feedback

This bit is meaningful only if you enable the loss-of-feedback detection feature by setting bit 15 of the most significant home position word of the parameter block.

If loss-of-feedback is enabled, and the servo positioning assembly detects a loss of feedback, it turns on the loss-of-feedback bit in the status word. If this bit is on, then the immediate stop bit in the status block is on, indicating that the 1771-ES expander has executed an immediate stop after detecting the loss-of-feedback.

Bit 17 Excess Error

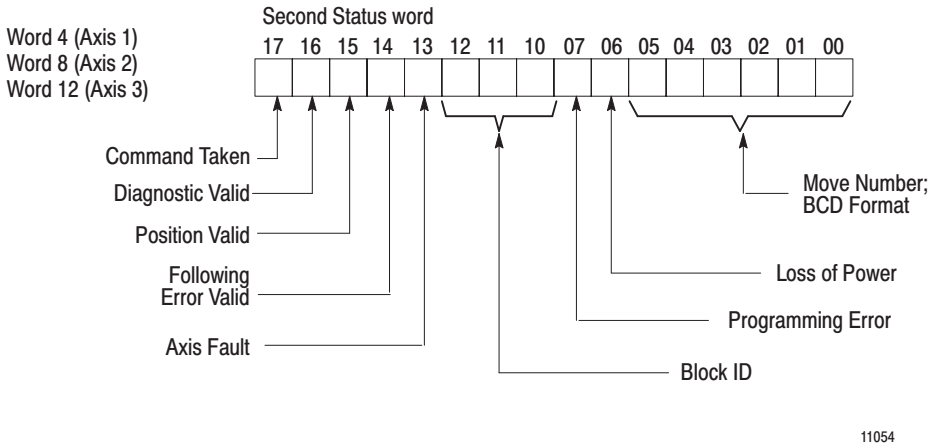
If following error equals or exceeds the excess following error value you enter in the parameter block, the 1771-M3 controller turns on this bit. Since excess following error turns on immediate stop, the immediate stop bit in the status block is also on. Additionally, if the 1771-ES expander applies feedrate reduction to an axis for which excess error is greater than the 106.25% built-in excess error value, then the feedrate reduction bit (bit 12) of the first status word for the axis is on. If, however, the excess error point you enter is less than 106.25%, then the feedrate reduction bit is not on.

Second Status Word

The second status word (Figure 7.5) identifies the active moveset and move as well as providing additional status bits.

CAUTION: The function of bits 06, 16, and 17 are different from the function of the corresponding bits for the series A servo positioning assembly. If you replace a series A assembly with a series B assembly without changing your program accordingly, you may cause unexpected results.

Figure 7.5
Second Status Word



Bit 0-5 Move Number

These bits indicate the active move within the moveset in BCD format.

Bit 6 Loss of Power

When set, this bit indicates a loss of power across one of the following sets of terminals:

- terminals 1 and 12 (input supply) of the left wiring arm
- terminals 1 and 6 (analog supply) of the right wiring arm

If this bit is on, then the immediate stop bit in the status block is on, indicating that immediate stop has been executed after detection of the loss of power.

Bit 7 Programming Error

If the 1771-M3 controller detects an illegal bit combination, such as a non-BCD value where one is expected, or an illegal bit combination in the command block, it turns on the programming error bit.

When this bit is on, bits 10 thru 12 of this status word provide a code to identify the block containing the programming error.

When you detect that this bit is on, you may want to turn on bit 11 of axis control word 2 in the command block to display diagnostic status in the 3rd and 4th status words for the axis.

Bits 12, 11, 10 Block ID

These bits are the block ID of the moveset block currently being executed, unless the diagnostic valid bit (bit 6) is on. When the programming error bit is on, bits 10 thru 12 indicate the block in which the error was detected:

ID (Bits 12, 11, 10)	Block
000	Parameter
001	Axis 1 Odd Moveset
010	Axis 2 Odd Moveset
011	Axis 3 Odd Moveset
100	Axis 1 Even Moveset
101	Axis 2 Even Moveset
110	Axis 3 Even Moveset
111	Command

Bit 13 Axis Fault

The 1771-M3 controller turns on this bit when communication between it and the 1771-ES expander is lost.

Bit 14 Following Error Valid

This bit is on if the next two status block words for this axis contain axis following error.

Bit 15 Position Valid

This bit is on if the next two status block words for this axis contain axis position.

If the axis position value exceeds the maximum allowable value (999.9999 in or 19999.999 mm), the servo positioning assembly turns off both the position valid and following error valid bits (bits 15 and 14), and sets the position value in the status block at the maximum value.

Bit 16 Diagnostic Valid

When you turn on the select diagnostic bit of axis control word 2 of the command block, this bit goes on to indicate that the position (or following error) words in the status block contain diagnostic information.

Bit 17 Command Taken

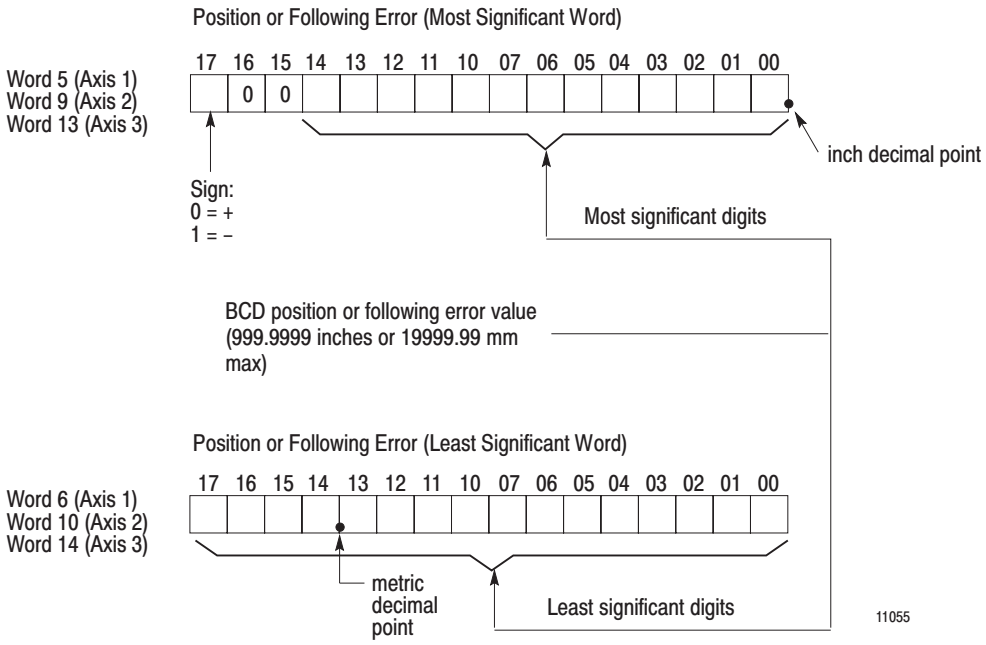
When you turn on the new-parameter, moveset override, offset preset, or get-new-preset-value bit in the command block, this bit goes on to indicate that the command has been taken. When you detect this bit to be on, you can turn off the command-block bit.

Position/Following-Error/Diagnostic Words

The 3rd and 4th status words for an axis provide either current axis position, following error, or diagnostic information. You can select which status to display by controlling the state of bits 11 and 15 of the axis control word 2 of the command block (refer to Figure 7.44 and its associated text for more information on Axis Control Word 2).

Turn off bits 11 and 15 to display the current axis position as shown in Figure 7.6. The maximum value is 999.9999 inch or 19999.999 mm. If the axis exceeds the maximum, it displays the maximum, and the position-valid bit goes off.

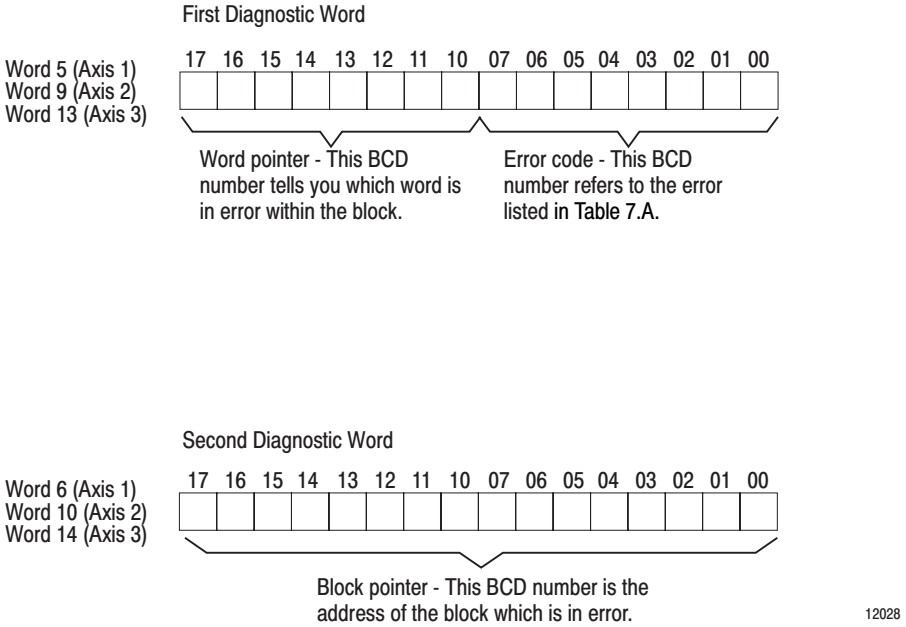
Figure 7.6
Position/ Following error/ Diagnostic Words - with Position or Following error Selected


Position or Following Error (Most Significant Word)
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 5 (Axis 1)
Word 9 (Axis 2)
Word 13 (Axis 3)
0 0
inch decimal point
Sign:
0 = +
1 = –
Most significant digits
BCD position or following error value
(999.9999 inches or 19999.99 mm
max)
Position or Following Error (Least Significant Word)
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 6 (Axis 1)
Word 10 (Axis 2)
Word 14 (Axis 3)
metric
decimal
point
Least significant digits
11055

Turn off bit 11 and turn on bit 15 to display the following error as shown in Figure 7.6. The maximum value is 999.9999 inch or 19999.999 mm. If the axis exceeds the maximum, it displays the maximum.

Turn on bit 11 to display the diagnostic status as shown in Figure 7.7.

Figure 7.7
Position/Following-Error/Diagnositc Words with Diagnostic Selected


First Diagnostic Word
Word 5 (Axis 1)
Word 9 (Axis 2)
Word 13 (Axis 3)
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word pointer - This BCD number tells you which word is in error within the block.
Error code - This BCD number refers to the error listed in Table 7.A.
Second Diagnostic Word
Word 6 (Axis 1)
Word 10 (Axis 2)
Word 14 (Axis 3)
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Block pointer - This BCD number is the address of the block which is in error.
12028

Also, this diagnostic status displays automatically when the 1771-M3 controller detects an error in the parameter block immediately after power-up or an invalid ID in a command block. The diagnostic status displays automatically in that case because the error prevents your selecting it through the command block.

The second diagnostic word is the block pointer. The block pointer is a BCD number that indicates the starting address of the block in error. The 1771-M3 controller gets these block pointers you enter into the parameter block or the moveset block.

The high byte (bit 10 thru 17) of the first diagnostic word is the word pointer. The word pointer is a BCD number (1 thru 64) that indicates which word is in error within the block.

The low byte (bits 00 thru 10) of the first diagnostic word is the error code. The error code is a BCD number that references the errors listed in table 7.A.

Use the block pointer and word pointer to identify the location of the problem. Then use the error code to determine the nature of the problem.

Table 7.A
Diagnostic Code Definitions

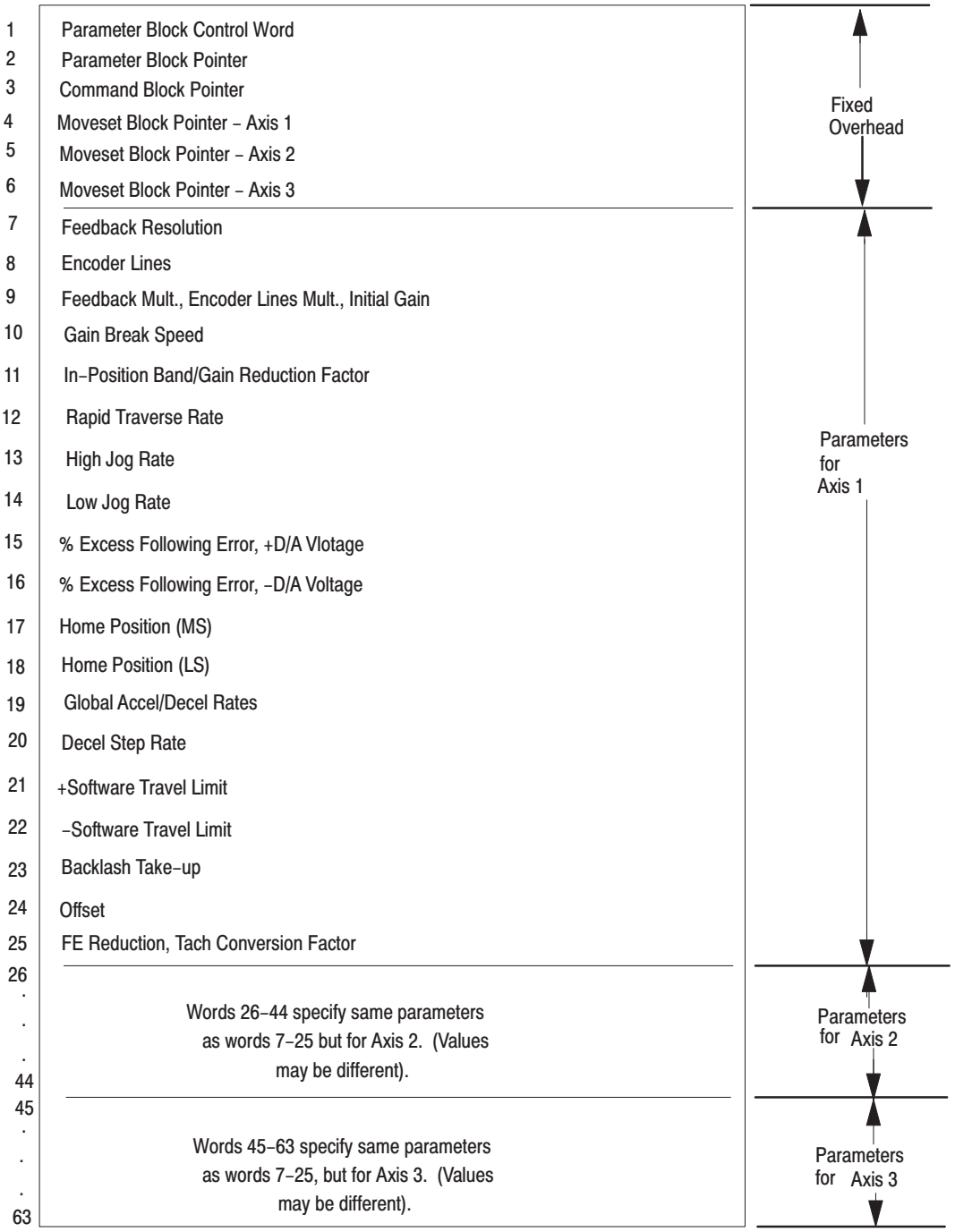
Code	Definition
01	Invalid block identifier.
02	Non-BCD number entered.
03	Invalid bit setting unused bits must be zero.
04	MS “metric only” bit set in inch format.
05	Overflow: Converted data is too large for internal registers.
06	Can only change feedback multiplier from a power-up rest.
07	Invalid “axes used” programmed.
08	Invalid write block address points.
09	Invalid feedback resolution (<0.00001 in. or 0.0001 mm).
10	Invalid feedback multiplier bit setting.
11	(Counts per rev) x (feedback mult) x (encoder lines mult)>32767 decimal.
12	D/A voltage too small for selected rapid rate.
13	Initial gain too small for selected rapid rate.
14	Rapid rate entered exceeds 250 kHz maximum input frequency.
15	Rapid rate entered exceeds 1/2 revolution of encoder/2.4ms.
16	Programmed velocity >rapid rate.
17	Invalid velocity exponent programmed.
18	Entered speed is too small for selected feedback resolution.
19	Accel velocity or decel value is too small for selected feedback resolution
20	Not as many valid SMCWs as there were moves declared in the MCW.
21	Local parameters or run at velocity not allowed for a preset or dwell.
22	Invalid preset position (must be an absolute position).
23	Invalid dwell time (must be≥20ms).
24	Escape move block can only have 1 move declared.
25	Invalid escape move block; only moveset blocks identified in the parameter block can be escape move blocks.
26	Cannot program a preset or dwell as an escape move.
27	A valid next-moveset pointer could not be found.
28	Command results in overflow of offset accumulator.
29	Attempted context switch with dual meaning bits on.
30	Attempted context switch while axis is commanding motion.
31	Manual mode only bit(s) on while in auto mode.

Code	Definition
32	Invalid motion command bit combination or command not allowed.
33	Invalid command (cannot process new parameters, preset, or offset commands while axis is in motion).
34	Attempted switch to auto mode before first marker is found.

Parameter Block

Through the parameter block you specify axis parameters such as software travel limits, home position value, servo gain; and rapid traverse rate. You specify these parameters for each axis individually (Figure 7.8).

Figure 7.8
Parameter Block - Showing Word Assignments


1 Parameter Block Control Word
2 Parameter Block Pointer
3 Command Block Pointer
4 Moveset Block Pointer – Axis 1
5 Moveset Block Pointer – Axis 2
6 Moveset Block Pointer – Axis 3
7 Feedback Resolution
8 Encoder Lines
9 Feedback Mult., Encoder Lines Mult., Initial Gain
10 Gain Break Speed
11 In–Position Band/Gain Reduction Factor
12 Rapid Traverse Rate
13 High Jog Rate
14 Low Jog Rate
15 % Excess Following Error, +D/A Vlotage
16 % Excess Following Error, –D/A Voltage
17 Home Position (MS)
18 Home Position (LS)
19 Global Accel/Decel Rates
20 Decel Step Rate
21 +Software Travel Limit
22 –Software Travel Limit
23 Backlash Take–up
24 Offset
25 FE Reduction, Tach Conversion Factor
26
Words 26–44 specify same parameters as words 7–25 but for Axis 2. (Values may be different).
44
45
Words 45–63 specify same parameters as words 7–25, but for Axis 3. (Values may be different).
63
Fixed Overhead
Parameters for Axis 1
Parameters for Axis 2
Parameters for Axis 3

The size of the parameter block you must provide depends on the number of axes:

Number of Axes	Size of Parameter Block
1	25 words
2	44 words
3	63 words

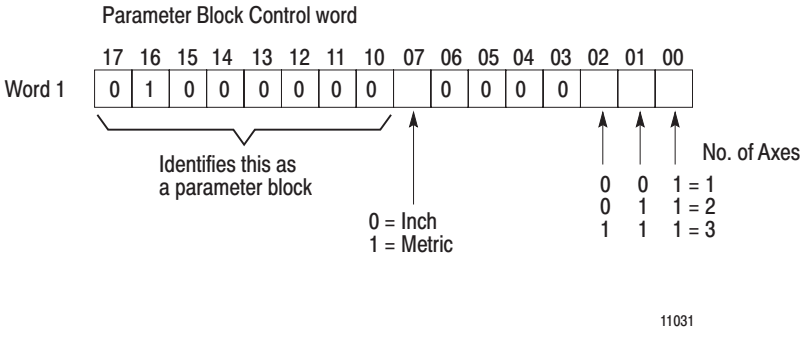
Your program must transfer the parameter block at power-up. After that, the 1771-M3 controller calls for your program to send the parameter block again only when you issue a new parameter or reset command.

Parameter Control Word

The parameter control word (Figure 7.9):

- identifies the block as the parameter lock (bits 10-17)
- specifies the units as either inch or metric (bit 7)
- identifies the number of axes in the system (bit 0,1, and 2)

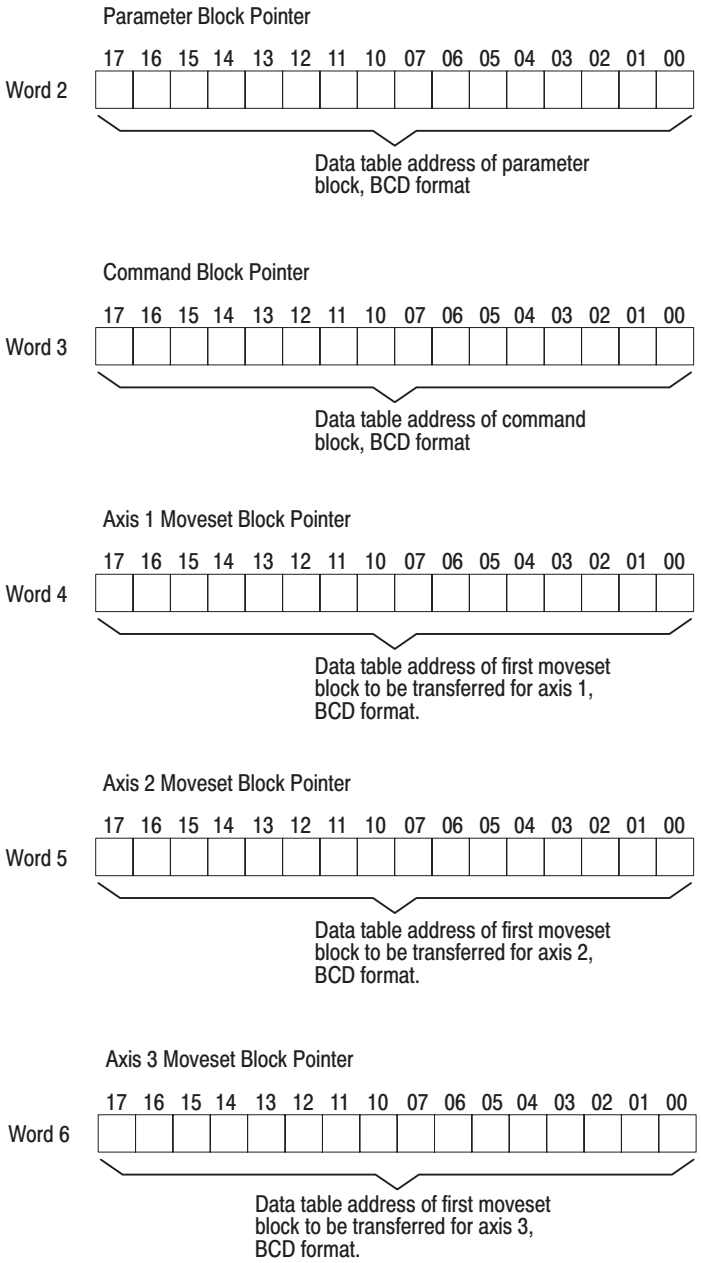
Figure 7.9
Parameter Block Control Word


Parameter Block Control word
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 1
0 1 0 0 0 0 0 0 0 0 0 0
Identifies this as
a parameter block
0 = Inch
1 = Metric
No. of Axes
0 0 1 = 1
0 1 1 = 2
1 1 1 = 3
11031

Address Pointers

Words 2 through 5 specify the starting addresses of the parameter, command, and first moveset block for each axis (Figure 7.10).

Figure 7.10
Address Pointer Words


Parameter Block Pointer
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 2
Data table address of parameter block, BCD format
Command Block Pointer
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 3
Data table address of command block, BCD format
Axis 1 Moveset Block Pointer
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 4
Data table address of first moveset block to be transferred for axis 1, BCD format.
Axis 2 Moveset Block Pointer
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 5
Data table address of first moveset block to be transferred for axis 2, BCD format.
Axis 3 Moveset Block Pointer
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 6
Data table address of first moveset block to be transferred for axis 3, BCD format.
11032

Important: The address pointer value you enter in each of these words must be a **BCD** value other than **000**.

For each axis, include only the address of the first moveset block in the parameter block. Include address pointers to subsequent movesets in the blocks that precede them. If you program an escape move, you must enter its address in the parameter block moveset address pointer word. You must do this because the escape move must be the first moveset transferred to the 1771-M3 controller, even though it is not the first moveset normally executed.

Feedback Resolution

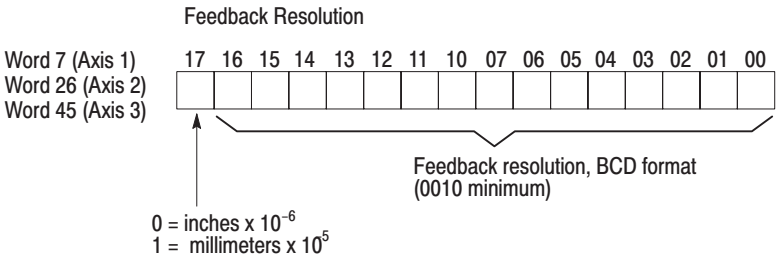
Feedback resolution is the smallest unit of axis motion that can be distinguished by the servo positioning assembly. That is, it is the distance the axis moves per feedback increment.

Enter the value of feedback resolution in the feedback resolution word of the parameter block (Figure 7.11). As described in chapter 3, feedback resolution is determined by the number of encoder lines, the feedback multiplier, and leadscrew pitch:

$$\text{Feedback Resolution} = \frac{\text{Axis Displacement per Encoder Rev.}}{(\text{Encoder Lines})\ (\text{Feedback Multiplier})}$$

If the system has no gearing, the axis displacement per revolution is the same as the leadscrew pitch or lead.

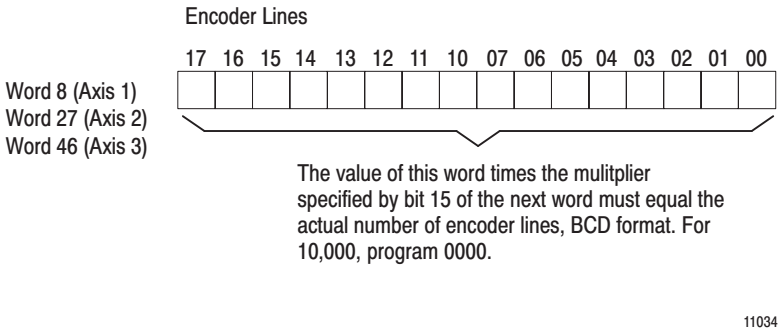
Figure 7.11
Feedback Resolution Word


Feedback Resolution
Word 7 (Axis 1)
Word 26 (Axis 2)
Word 45 (Axis 3)
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Feedback resolution, BCD format
(0010 minimum)
0 = inches x 10-6
1 = millimeters x 10^5
11033

Encoder Lines

This word specifies the number of encoder lines per encoder revolution (Figure 7.12).

Figure 7.12
Encoder Lines Word


Encoder Lines
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 8 (Axis 1)
Word 27 (Axis 2)
Word 46 (Axis 3)
The value of this word times the mulitplier specified by bit 15 of the next word must equal the actual number of encoder lines, BCD format. For 10,000, program 0000.
11034

The value of this word times the encoder lines multiplier specified by bit 15 of the **next** higher word, must equal the actual number of lines on the encoder.

You can enter values up to 10,000 with the x1 multiplier. Entering zero (0000) specifies 10,000 lines. You can enter higher values by using the x4 multiplier.

For example, if your encoder has 12,000 lines, you can enter 3000 in the encoder-lines word, and turn on bit 15 of the next word to indicate x4: 3,000 x 4 = 12,000 lines.

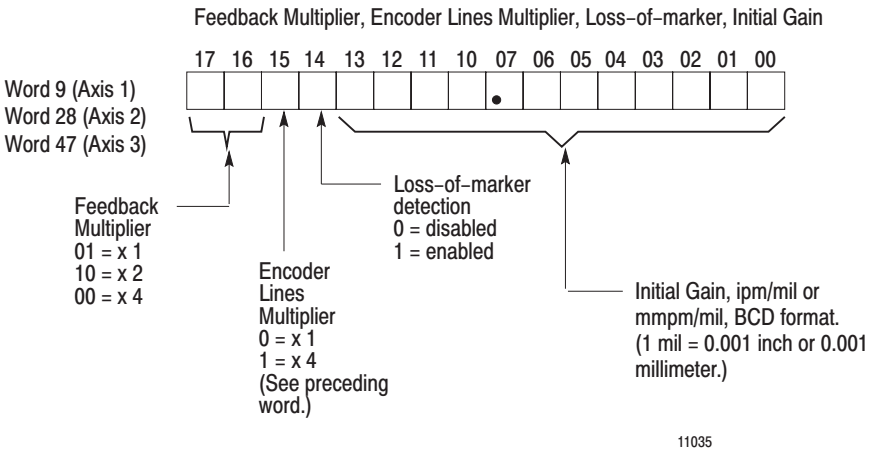
The status block indicates a programming error after transfer of the parameter block if:

(Encoder Lines) x (Feedback Multiplier) x (Encoder Lines Multiplier) > 32,767

Initial-Gain/Multipliers

Bits 0 thru 13 of this word (Figure 7.13) specify the servo gain for this axis at speeds below the gain break speed specified in word 10. You must enter gain values in BCD format, from 0.01 to 9.99 ipm/mil or from 0.01 to 9.99 mmpm/mil. (A mil is 0.001 inch or 0.001 mm.)

Figure 7.13
Feedrate Multiplier, Encoder Lines Multiplier, and Initial Gain Word


Feedback Multiplier, Encoder Lines Multiplier, Loss–of–marker, Initial Gain
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 9 (Axis 1)
Word 28 (Axis 2)
Word 47 (Axis 3)
Feedback Multiplier
01 = x 1
10 = x 2
00 = x 4
Encoder Lines Multiplier
0 = x 1
1 = x 4
(See preceding word.)
Loss–of–marker detection
0 = disabled
1 = enabled
Initial Gain, ipm/mil or mmpm/mil, BCD format.
(1 mil = 0.001 inch or 0.001 millimeter.)
11035

Servo gain is the ratio of axis speed to following error:

$$\text{Gain} = \frac{\text{Axis Speed}}{\text{Following Error}}$$

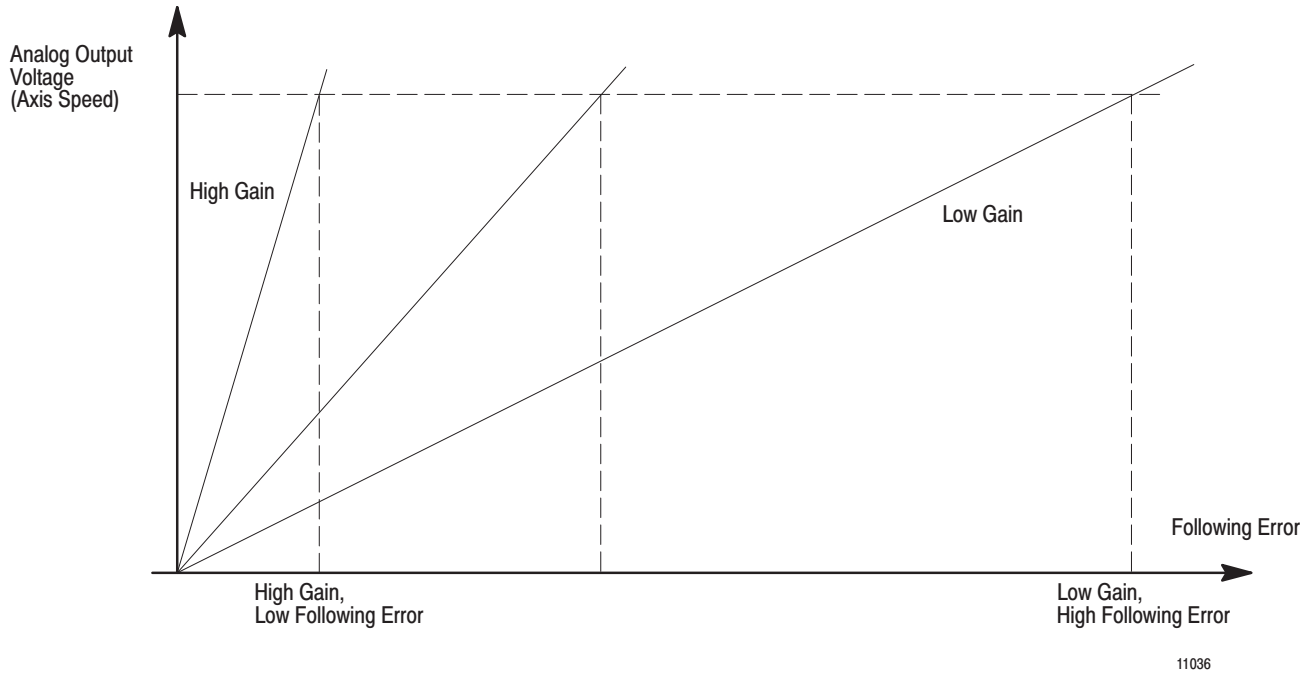
Following error is the difference between the axis position commanded by the servo expander and the actual axis position indicated by encoder feedback.

Servo gain affects axis response to positioning commands from the 1771-ES expander module. Figure 7.14 shows how different gain values affect system responsiveness. The horizontal axis represents following error. The vertical axis represents analog output voltage. Since analog output voltage is directly proportional to axis speed, you can use the vertical axis to represent either variable.

If gain is relatively high, following error will be relatively small, because the system will be more sensitive to changes in following error. If gain is

low, following error becomes relatively larger, because the system is not as responsive to changes in following error. Choose a gain value to match the capability of your axis drives, motors, and mechanics, and provide adequate system response.

Figure 7.14
Following Error Vs. Speed for Various Gains



Parameter block values for gain and in-position band must provide a stable system and maintain desired positioning accuracy. If gain is too high, the axis may overshoot programmed endpoints and oscillate, or "hunt," about them. If gain is too low, the axis may stop before it is within the desired in-position band. You can increase in-position band, but this decreases positioning accuracy.

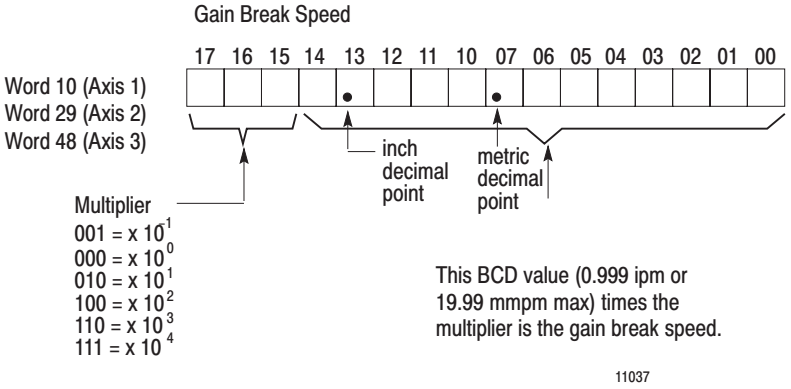
Use bit 15 of this word to select the encoder lines multiplier. This encoder lines multiplier you select, times the encoder lines value you select in the previous word, must match the number of lines per revolution of the encoder.

Use bits 16 and 17 of this word to select the feedback multiplier. The feedback multiplier you select affects the value you must enter for the feedback resolution word.

Gain Break Speed

At axis speed below the gain break value you enter into the gain break word (Figure 7.15), servo gain is the initial gain programmed in the preceding word.

Figure 7.15
Gain Break Speed Word


Gain Break Speed
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 10 (Axis 1)
Word 29 (Axis 2)
Word 48 (Axis 3)
inch decimal point
metric decimal point
Multiplier
001 = x 10^1
000 = x 10^0
010 = x 10^1
100 = x 10^2
110 = x 10^3
111 = x 10^4
This BCD value (0.999 ipm or 19.99 mmpm max) times the multiplier is the gain break speed.
11037

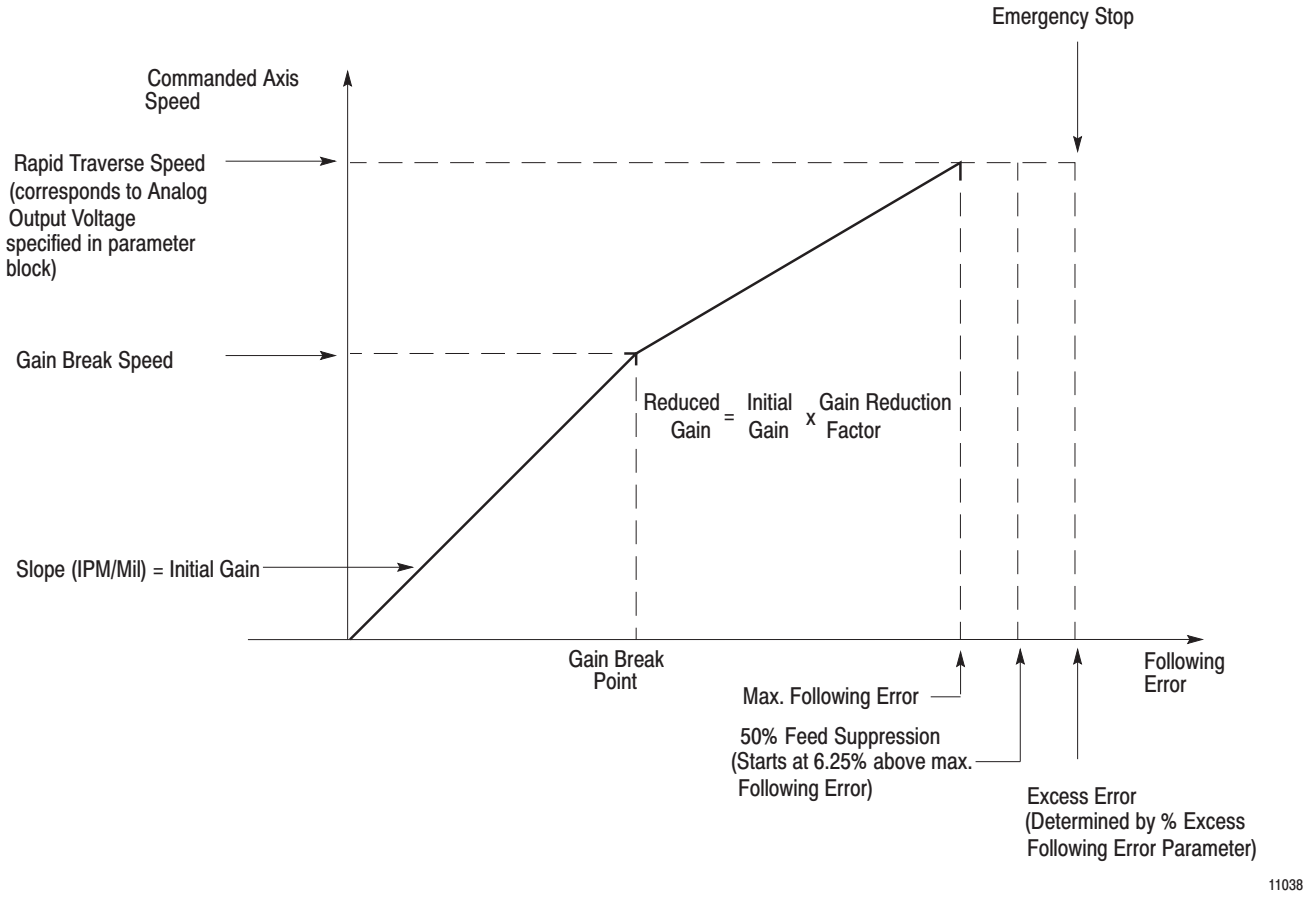
At speeds equal to and above the gain break value you enter into this word, the servo positioning assembly reduces servo gain by the gain reduction factor specified in the next word of the parameter block.

The gain break plot of Figure 7.16 illustrates the concept of gain break.

Typically, gain at axis speeds below the gain break velocity is relatively high to allow precise axis positioning. Reduced gain at axis speeds above gain break velocity allows for better stability at higher axis speeds.

Gain break velocity can be no greater than rapid traverse rate. If there is to be no gain break point for an axis, program the rapid traverse speed in this word.

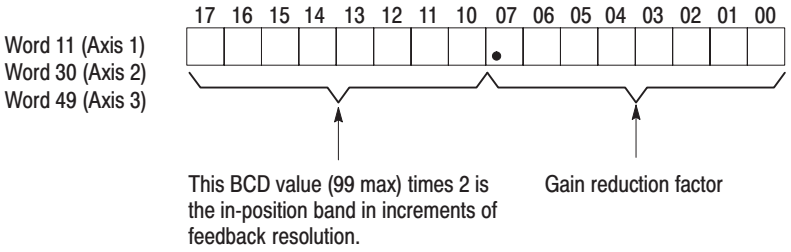
Figure 7.16
Gain Break Plot


Emergency Stop
Commanded Axis Speed
Rapid Traverse Speed (corresponds to Analog Output Voltage specified in parameter block)
Gain Break Speed
Reduced Gain = Initial Gain x Gain Reduction Factor
Slope (IPM/Mil) = Initial Gain
Gain Break Point
Following Error
Max. Following Error
50% Feed Suppression (Starts at 6.25% above max. Following Error)
Excess Error (Determined by % Excess Following Error Parameter)
11038

Gain-Reduction Factor

Bits 0-7 of the in-position-band and gain-break-factor word (Figure 7.17) specify the gain reduction factor. The initial gain of the axis is multiplied by this factor to obtain the reduced gain value for axis speeds above the gain break speed.

Figure 7.17
In-position Band, Gain Reduction Factor Word


17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 11 (Axis 1)
Word 30 (Axis 2)
Word 49 (Axis 3)
This BCD value (99 max) times 2 is the in-position band in increments of feedback resolution.
Gain reduction factor
11039

$$\text{Gain Reduction Factor} = \frac{\text{Reduced Gain}}{\text{Initial Gain}}$$

For example, if the initial gain is one, and you want the reduced gain to be 0.5, program .50 as the value for gain reduction factor.

$$\text{Gain Reduction Factor} = \frac{.5}{1} = .50$$

The gain reduction factor must be less than 1.0. If you program zero, the system gain for any axis speed will be the initial gain. If gain break velocity is zero and you program a non-zero gain reduction factor, system gain for any axis speed is the initial gain times the gain reduction factor. Enter the gain reduction factor in BCD format.

In-Position Band

The size of the in-position band is measured in increments of the feedback resolution of the axis. Program a 2-digit BCD value that is half the desired in-position band in bits 10-17 of the in-position-band and gain-break-factor word (Figure 7.17). If you program zero as the in-position band parameter value, the servo positioning assembly automatically makes the active in-position band ± 2 feedback increments.

The 1771-M3 controller turns on the in-position bit when the done bit is on in the status block and the axis is within the in-position band.

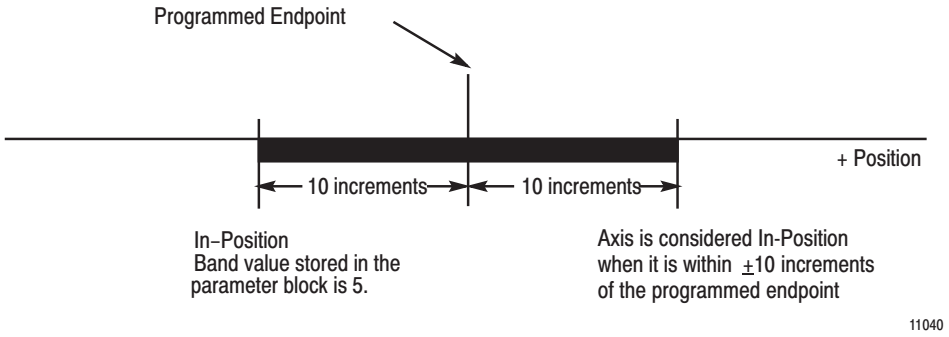
The axis must be in-position before the following actions can take place:

- Manual mode commands are not executed unless the in-position bit is on.
- In auto mode, the start command is not executed unless the in-position bit in the status block is on.

- When the direction of axis motion is reversed, the in-position bit in the status block must be on before axis motion in the reverse direction can occur.

Note that the value you enter for the in-position band is actually half the desired in-position band value. For example, if the in-position band value you enter is 5, then the servo positioning assembly considers the axis in-position if it is within ±10 increments of feedback resolution of the programmed endpoint. Figure 7.18 illustrates the concept of in-position band.

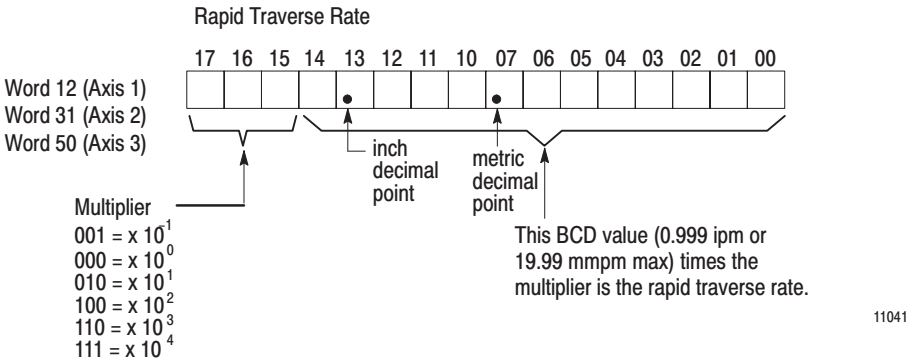
Figure 7.18
In-position Band Example



Rapid Traverse Rate

The rapid traverse rate you enter (Figure 7.19) is the highest feedrate the axis can attain. It is associated with open travel of the axis. The servo positioning assembly uses this rate for the go home operation and for moves that you program to use the global feedrate.

Figure 7.19
Rapid Traverse Rate Word


Rapid Traverse Rate
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 12 (Axis 1)
Word 31 (Axis 2)
Word 50 (Axis 3)
inch decimal point
metric decimal point
Multiplier
001 = x 10^1
000 = x 10^0
010 = x 10^1
100 = x 10^2
110 = x 10^3
111 = x 10^4
This BCD value (0.999 ipm or 19.99 mmpm max) times the multiplier is the rapid traverse rate.
11041

The rapid traverse rate is limited by several parameters. The servo positioning assembly detects a programming error and inhibits axis motion when you enter a rapid traverse rate that violates any of the following formulas:

$$RR < \frac{12{,}500 \times FR \times FM \times EL}{1.28}$$ (Table 7.A, code 15)

$$RR < \frac{1.5 \times 10^7 \times FR \times FM}{1.28}$$ (Table 7.A, code 14)

$$RR < 4 \times 10^6 \times FR \times D/A$$ (Table 7.A, code 12)

$$RR < \frac{6.5 \times 10^7 \times FR \times IG}{1.28}$$ (Table 7.A, code 13)

Where the following are parameters you enter:

RR = rapid traverse rate

FR = feedback resolution

FM = feedback multiplier (1, 2, or 4)

EL = encoder lines per revolution

D/A = maximum D/A voltage

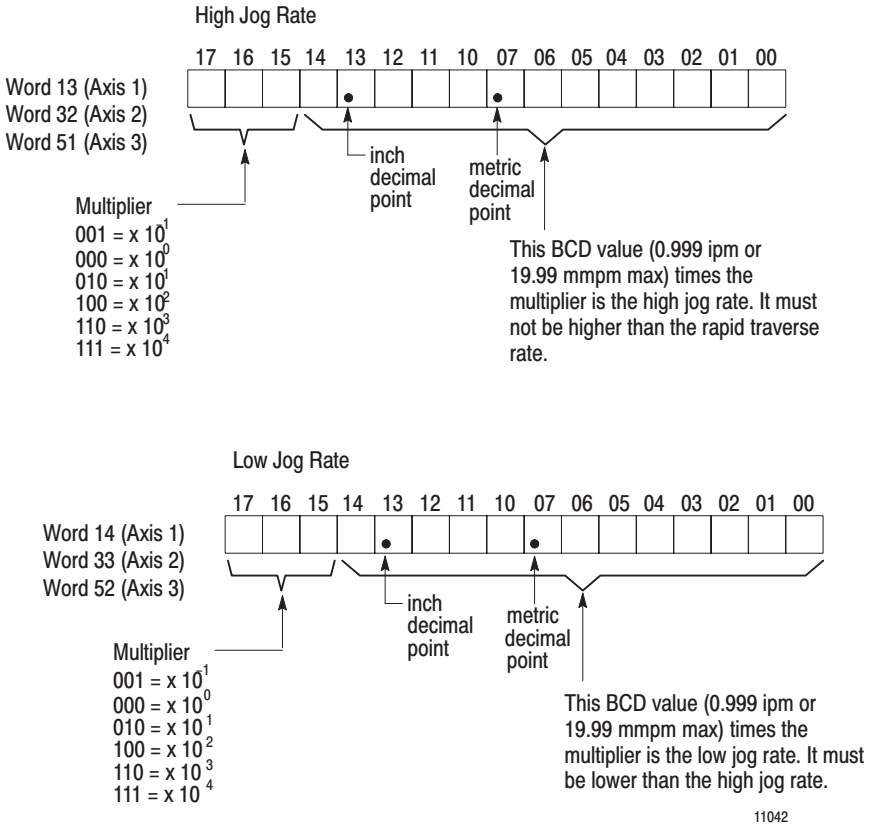
IG = initial gain

These formulas include an allowance for a 127% feedrate override factor. These formulas apply to both ipm and mmpm.

Jog Rate

The high and log jog rate words (Figure 7.20) specify the speeds at which you can jog the axis. You can jog the axis only in manual mode. Program the values in BCD format. The operator can select jog speed (high or low) by controlling the jog rate select bit in the command block.

Figure 7.20
Jog Rate Words

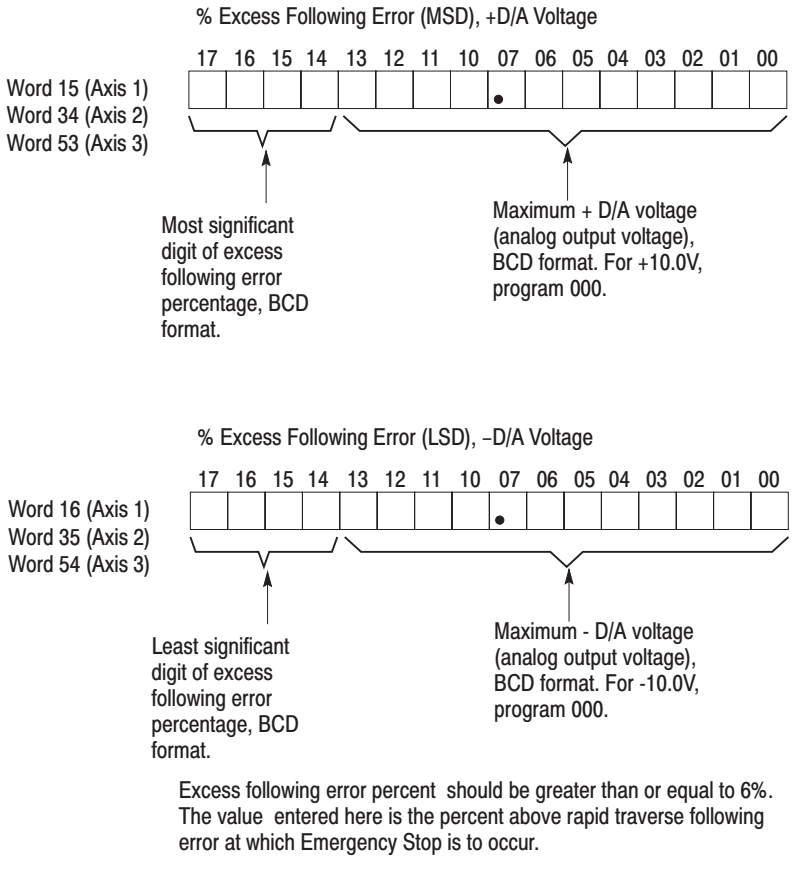

High Jog Rate
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 13 (Axis 1)
Word 32 (Axis 2)
Word 51 (Axis 3)
inch decimal point
metric decimal point
Multiplier
001 = x 10^1
000 = x 10^0
010 = x 10^1
100 = x 10^2
110 = x 10^3
111 = x 10^4
This BCD value (0.999 ipm or 19.99 mmpm max) times the multiplier is the high jog rate. It must not be higher than the rapid traverse rate.
Low Jog Rate
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 14 (Axis 1)
Word 33 (Axis 2)
Word 52 (Axis 3)
inch decimal point
metric decimal point
Multiplier
001 = x 10^1
000 = x 10^0
010 = x 10^1
100 = x 10^2
110 = x 10^3
111 = x 10^4
This BCD value (0.999 ipm or 19.99 mmpm max) times the multiplier is the low jog rate. It must be lower than the high jog rate.
11042

% Excess Following Error

The % excess following error parameter is a 2-digit BCD number that the 1771-ES expander interprets as a percentage above the following error allowed at the rapid traverse rate. Programmable excess following error values can thus range from 0 through 99. Program the most significant digit in bits 14 through 17 of the first word, and the least significant digits in bits 14 through 17 of the second word (Figure 7.21).

This parameter specifies maximum allowable axis following error. When the following error reaches the maximum value permitted as specified by the % excess following error parameter, the servo positioning assembly stops axis motion by commanding immediate stop (Figure 7.16).

Figure 7.21
Excess Following Error, D/A Voltage Words


% Excess Following Error (MSD), +D/A Voltage
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 15 (Axis 1)
Word 34 (Axis 2)
Word 53 (Axis 3)
Most significant digit of excess following error percentage, BCD format.
Maximum + D/A voltage (analog output voltage), BCD format. For +10.0V, program 000.
% Excess Following Error (LSD), –D/A Voltage
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 16 (Axis 1)
Word 35 (Axis 2)
Word 54 (Axis 3)
Least significant digit of excess following error percentage, BCD format.
Maximum - D/A voltage (analog output voltage), BCD format. For -10.0V, program 000.
Excess following error percent should be greater than or equal to 6%. The value entered here is the percent above rapid traverse following error at which Emergency Stop is to occur.
11043

Feedrate Reduction

When axis following error reaches 106.25% of rapid traverse following error, the servo positioning assembly automatically reduces feedrate by 50% of the feedrate value. This feedrate reduction provides an opportunity for following error to decrease. Feedrate returns to the programmed value when following error is reduced to less than or equal to 106.25% of rapid traverse following error and the current move is completed.

Note that if the excess following error value you enter is less than or equal to 6%, the axis executes immediate stop before following error reaches 106.25% of rapid traverse following error.

+ and - D/A Voltage

Bits 00 thru 13 in words 15 and 16 (34 and 35, 53 and 54) specify the maximum servo output voltage that is available to command rapid traverse feedrate in the positive and negative directions (Figure 7.21).

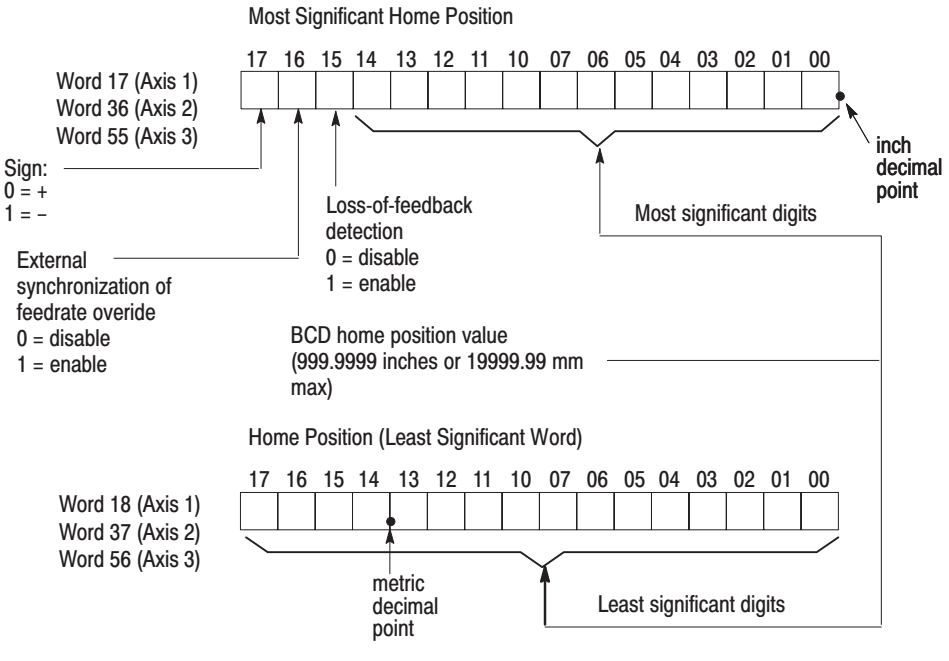
Enter values for these parameters in BCD format in the range of 0.01V to 9.99V. Programming 0 causes the D/A voltage value to default to 10V.

Initially set them to the maximum value the servo drive will accept. The plus and minus D/A voltage values needn't be equal. You can enter them as different values to compensate for directional differences in drive performance during axis integration (chapter 9).

Home Position Value

Words 17 and 18 (36 and 37, 55 and 56) specify the axis home position value (Figure 7.22). Bits 0 through 14 of the first word contain the most significant digits. Bit 17 of the first word specifies the sign of the home position value. When the servo positioning assembly performs a search home or initialize home operation, it sets the axis position register to the value you enter for this parameter.

Figure 7.22
Home Position Words



Loss-of-Feedback Detection Enable

Turn bit 15 off until you complete the open-loop and closed-loop axis integration procedures (chapter 9).

Then turn on bit 15 of the most significant home position word to enable the loss-of-feedback detection feature of the 1771-ES expander.

WARNING: Once you have completed the axis integration procedures, **never** turn this bit **off**. Without loss-of-feedback detection, if encoder or tachometer feedback is lost, unexpected axis motion can occur, resulting in damage to equipment and/or injury to personnel.

External Synchronization of Feedrate Override

Turn on bit 16 of the most significant home position word to have the 1771-ES expander recognize the feedrate override enable input (Figure 7.22).

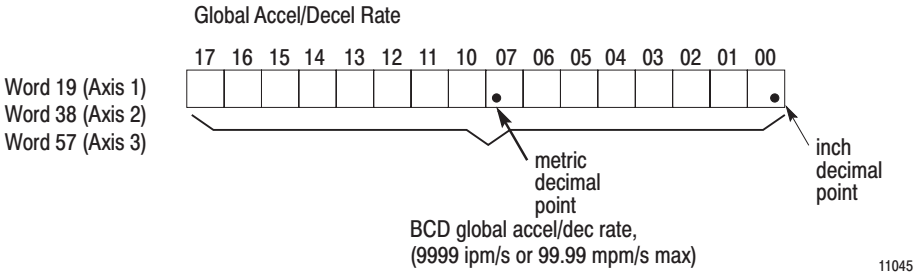
With this bit off, you change the feedrate by the percentage you enter in the command block when you enable feedrate override in the command block. However, you change the feedrate for a particular move only if you had enabled feedrate override in the move block.

With the bit on, you must still enable feedrate override in the command block and move block before feedrate changes. However, the 1771-ES expander will not change the feedrate until you close the feedrate-override-enable input. This allows you to synchronize the feedrate override of several axes.

Global Accel/Decel Rate

Word 19 (38, 57) specifies the acceleration and deceleration rate the servo positioning assembly uses for all jogs and for moves in movesets for which you do not enter local acceleration and deceleration rates. It is also the deceleration value used when executing a slide stop during manual mode operation of an axis or when you issue a reset command during axis motion. (Figure 7.23).

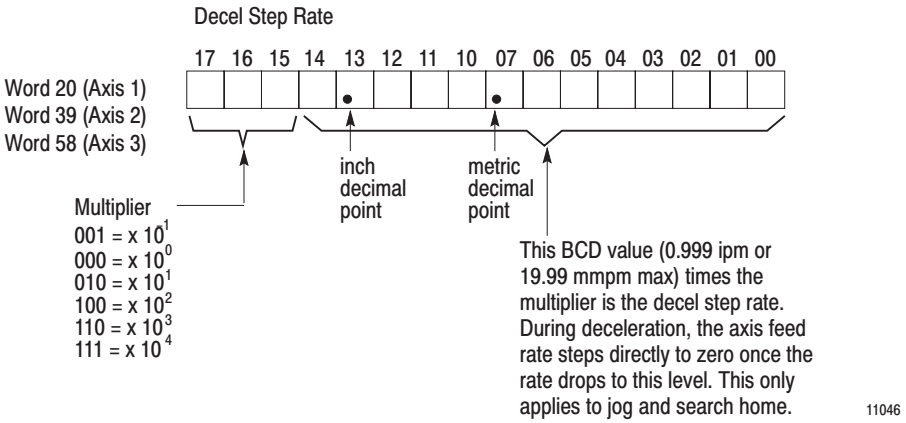
Figure 7.23
Global Accel/Decel Rate Word


Global Accel/Decel Rate
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 19 (Axis 1)
Word 38 (Axis 2)
Word 57 (Axis 3)
metric decimal point
inch decimal point
BCD global accel/dec rate,
(9999 ipm/s or 99.99 mpm/s max)
11045

Decel Step Rate

Word 20 (39, 58) specifies the deceleration step rate (Figure 7.24). This parameter applies to deceleration of axis motion when the servo positioning assembly is in the manual mode. At axis feed rates equal to or less than that specified by this word, the servo positioning assembly ignores the programmed deceleration rate, and steps axis feed rate directly to zero. This parameter is not effective in auto mode, and applies only to jog and search home operations.

Figure 7.24
Deceleration Step Rate Word


Decel Step Rate
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 20 (Axis 1)
Word 39 (Axis 2)
Word 58 (Axis 3)
Multiplier
001 = x 10^1
000 = x 10^0
010 = x 10^1
100 = x 10^2
110 = x 10^3
111 = x 10^4
inch decimal point
metric decimal point
This BCD value (0.999 ipm or 19.99 mmpm max) times the multiplier is the decel step rate. During deceleration, the axis feed rate steps directly to zero once the rate drops to this level. This only applies to jog and search home.
11046

Software Travel Limits

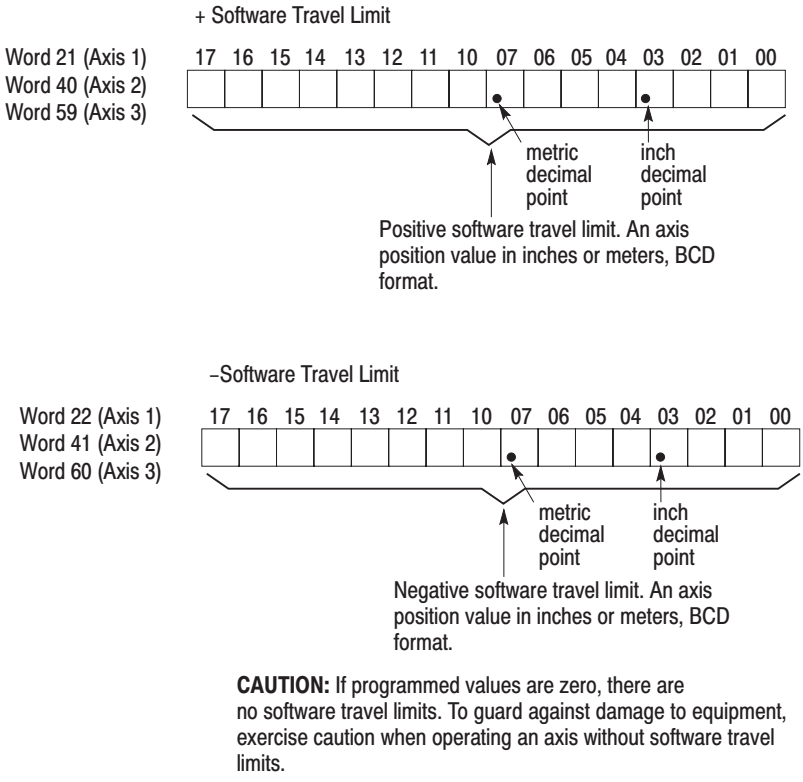
Words 21 and 22 (40 and 41, 59 and 60) specify the axis position values of the axis software travel limits (Figure 7.25). When a programmed move calls for the axis to move beyond a software travel limit, if there is time, the servo positioning assembly automatically decelerates the axis at the programmed rate for the current move so that it stops at or before the software travel limit position.

If there is no time to decelerate the axis before the limit is exceeded, the servo positioning assembly executes an immediate stop. This could occur following a continuous move because the next move starts with the feedrate of the previous move rather than zero.

Software travel limit values are axis position values. Note that if zero is the programmed travel limit value, there is no software travel limit. The absolute positions of the software travel limit vary with changes in axis position value due to preset or home commands. In addition to the software travel limit you must have extreme axis overtravel limit switches wired in the master control relay circuit.

CAUTION: If programmed values for software travel limits are zero, there are no software travel limits. To guard against damage to equipment, use caution when operating an axis without software travel limits.

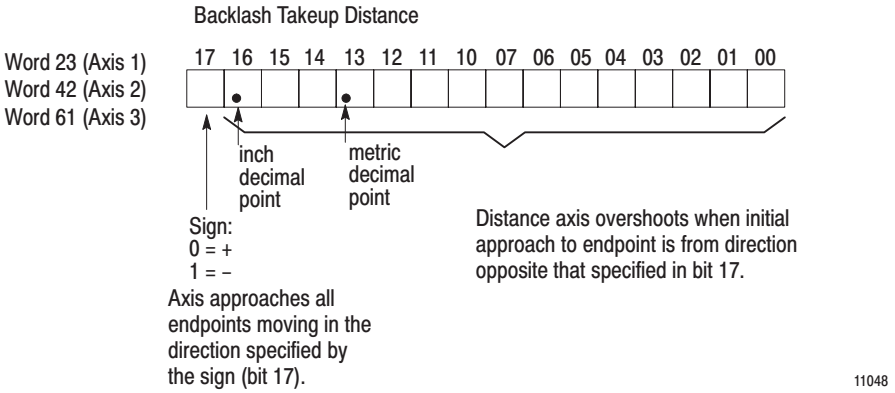
Figure 7.25
Software Travel Limit Words


+ Software Travel Limit
Word 21 (Axis 1)
Word 40 (Axis 2)
Word 59 (Axis 3)
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
metric decimal point
inch decimal point
Positive software travel limit. An axis position value in inches or meters, BCD format.
–Software Travel Limit
Word 22 (Axis 1)
Word 41 (Axis 2)
Word 60 (Axis 3)
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
metric decimal point
inch decimal point
Negative software travel limit. An axis position value in inches or meters, BCD format.
CAUTION: If programmed values are zero, there are no software travel limits. To guard against damage to equipment, exercise caution when operating an axis without software travel limits.
11047

Backlash Takeup

Backlash takeup helps minimize axis positioning inaccuracy caused by mechanical play in the axis positioning system. Word 23 (42, 61) is the backlash takeup word (Figure 7.26).

Figure 7.26
Backlash Takeup Word


Backlash Takeup Distance
Word 23 (Axis 1)
Word 42 (Axis 2)
Word 61 (Axis 3)
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
inch decimal point
metric decimal point
Sign:
0 = +
1 = –
Axis approaches all endpoints moving in the direction specified by the sign (bit 17).
Distance axis overshoots when initial approach to endpoint is from direction opposite that specified in bit 17.
11048

In this word, bit 17 specifies the direction the axis is to move in approaching all programmed endpoints. When the axis approaches an endpoint at which it is to stop while moving in the specified direction, it simply stops at the endpoint. If the axis approaches the endpoint from the opposite direction,it overshoots the endpoint by the amount you specify in bits 00 thru 16, then returns to the endpoint from the opposite direction.

Consider the example of your entering +.0010 in the backlash takeup word:

- If the axis is moving in the positive direction it stops at the programmed endpoint without overshoot.
- If the axis is moving in the negative direction, it overshoots the endpoint by 0.001 inch, then returns to the programmed endpoint.

Backlash takeup affects only halt moves that command the axis to stop at a move endpoint. For blended moves, backlash takeup has no effect. Also, backlash takeup is active only in auto mode. Backlash takeup has no effect on axis motion in the manual mode.

This parameter has a 4-digit BCD value in the range of 0.0001 to 0.7999 inches or 0.001 to 7.999 mm.

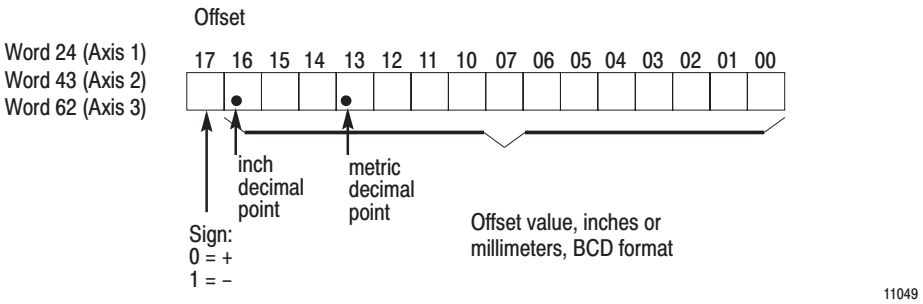
Offset

Word 24 (43, 62) specifies the value the servo positioning assembly adds to the offset accumulator when the servo positioning assembly executes one of the following:

- a position-with-offset move in a moveset
- an offset command from the command block

This parameter has a 4-digit BCD value that can be in the range ±0.0001 to ±0.7999 inches or ±0.001 to ±7.999 mm (Figure 7.27).

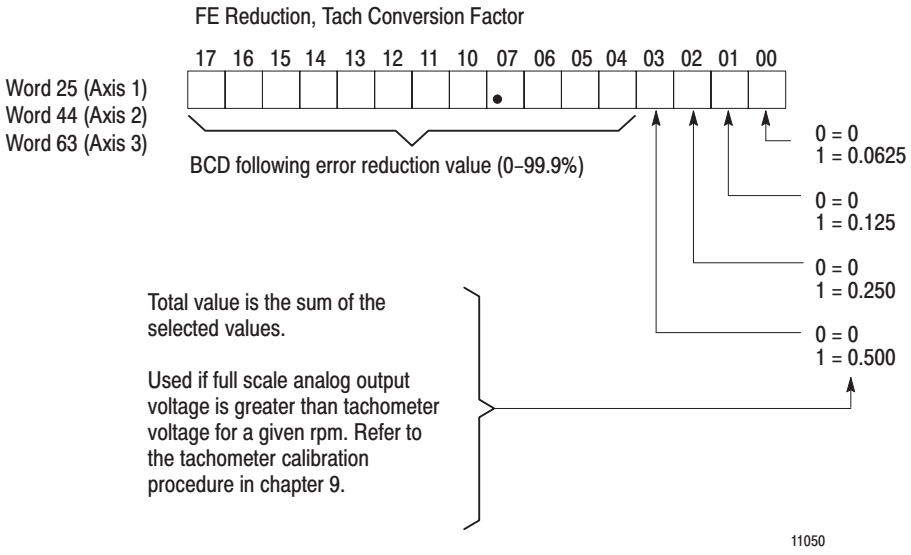
Figure 7.27
Offset Word


Offset
Word 24 (Axis 1)
Word 43 (Axis 2)
Word 62 (Axis 3)
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
inch decimal point
metric decimal point
Sign:
0 = +
1 = –
Offset value, inches or millimeters, BCD format
11049

Tachometer Conversion Factor

The servo positioning assembly uses bits 00 thru 03 of word 25 (44, 63) for its loss-of-feedback detection feature (Figure 7.28). The tachometer calibration procedure explains how to select the value for this word.

Figure 7.28
Following Error Reduction/Tachometer Conversion Factor Word


FE Reduction, Tach Conversion Factor
17 16 15 14 13 12 11 10 07 06 05 04 03 02 01 00
Word 25 (Axis 1)
Word 44 (Axis 2)
Word 63 (Axis 3)
BCD following error reduction value (0–99.9%)
0 = 0
1 = 0.0625
0 = 0
1 = 0.125
0 = 0
1 = 0.250
0 = 0
1 = 0.500
Total value is the sum of the selected values.
Used if full scale analog output voltage is greater than tachometer voltage for a given rpm. Refer to the tachometer calibration procedure in chapter 9.
11050

Each of bits 0 thru 3 corresponds to a factor. The total factor used by loss-of-feedback detection equals the sum of the individual factors selected.

If you are not using the loss-of-feedback detection feature, or if tachometer voltage is greater than or equal to 10V, program zero for all bits of this word.

Following Error Reduction

The servo positioning assembly accepts the BCD value you enter into bits 04 thru 17 as the following error reduction value (Figure 7.28).

You can command a reduction of the following error by 0 through 99.9%. The 1771-ES expander reduces the following error through feed forwarding without increasing the positioning loop gain.

Consider an example in which you have entered an initial gain value of 1.00. With an axis speed of 10 ipm, without following error reduction, the following error would be 10 mils.

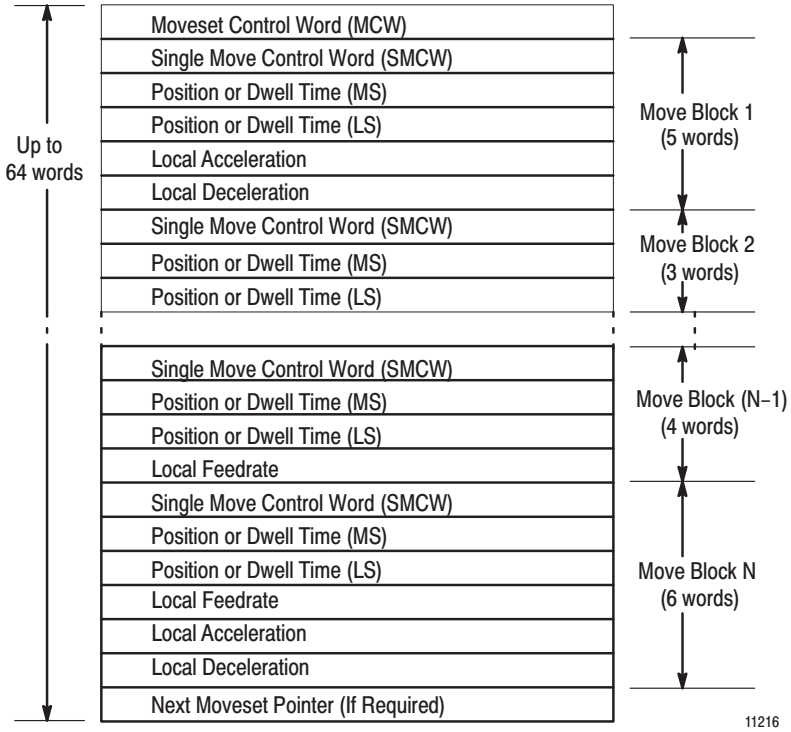
$$FE = \frac{\text{speed}}{\text{gain}} = \frac{10 \text{ ipm}}{1 \text{ ipm/mil}} = 10 \text{ mil s}$$

However, if you enter a following error reduction value of 70.0 the following error at 10 imp is reduced from 10 mils to 3 mils.

Moveset Block

A moveset block contains a number of move blocks through which it describes axis motion for a sequence of moves (Figure 7.29).

Figure 7.29
Moveset Block- Showing Word Assignments



Each move requires a move block of at least three words (a single move control word, and two words to define position or dwell time). A move block may have as many as six words (a single move control word, two position words, a rate word, an accel word, and a decel word). In addition, two words (moveset control word and next moveset pointer) apply to the entire moveset block. Since the moveset block may contain no more than 64 words, the largest possible number of moves a single

block can describe is 21. All 21 moves would have to use global accel/decel and final rate values.

Upon request from the status block, the PC processor writes a moveset block to the 1771-M3 controller, which transfers the move blocks to the 1771-ES expander one at a time. The servo expander generates analog voltage to command axis motion as programmed.

The first word of a moveset block is the moveset control word (MCW).

Following the MCW are the move blocks. Each move block consists of a single move control word (SMCW), two position (or dwell) words, and may contain words for local feedrate, accel rate, and decel rate, depending on whether you select local or global rates by the SMCW.

You can leave words of zeros before and after move blocks. This gives you flexibility. For example, you could remove a move block without changing the location of the other move blocks within the data table. However, you must respecify the number of moves in the moveset control word.

If the end of program bit is off, the last word in a moveset block must be an address pointer (in BCD format) to the moveset block that should be executed next.

Moveset Control Word

The MCW word identifies the block as a moveset block, indicates whether you program the axis in inch or metric units, specifies the number of moves in the moveset and whether or not the moveset defines an escape